University of Alberta

QUANTITATIVE APPROACHES FOR FAULT DETECTION AND DIAGNOSIS IN PROCESS INDUSTRIES

by

Harigopal Raghavan ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**

in

Process Control

Department of Chemical and Materials Engineering

Edmonton, Alberta
Fall 2004

Library and
Archives Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-96006-4
Our file Notre référence
ISBN: 0-612-96006-4

The author has granted a non-exclusive license allowing the Library and Archives Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Dedicated to

Amma, Appa, Akka and Athimber

for their love, support and faith

Acknowledgements

I thank my supervisor Prof. Sirish Shah, for his excellent guidance. The discussions I had with him during the execution of my thesis projects helped shape my research and bring the necessary depth and breadth to my thesis.

I am indebted to Dr. Weihua Li for initiation into the area of Fault diagnosis and Subspace Identification and Prof. Shankar Narasimhan for stimulating arguments on everything under the sun. I thank Prof. Masliyah for his suggestions related to the bitumen recovery project. I thank Prof. Kannan Moudgalya for suggestions on my thesis. I thank my academic and industrial co-workers, Arun, Bhushan, Johan, Ramesh, Rohit, Vikas, Weihua, Zhengang, Brian Doucette and Chris Robson for discussions during the course of our projects. I reserve special thanks to Vinay for interesting discussions on control theory.

I would like to thank the NSERC-Matrikon-ASRA Industrial Research Chair programme and industrial partners Matrikon, Suncor, Millar Western and Precarn for financial and industrial support.

I have been fortunate to work as a part of one of the best process control groups in the world. The high quality of control and statistics courses helped to get me started on my research with confidence. I thank Profs. Tongwen Chen, Peter Hooper, Biao Huang, Witold Pedrycz, Sirish Shah, Doug Wiens and Dr. Weihua Li for their high technical standards and dedication to teaching. I thank my colleagues in the computer process control group, Arun, Bhushan, Monjur, Raghu, Ramesh, Salim, Shoukat, Syed, Vikas, Vinay, Zhengang, Dr. Mani Bhushan and Prof. Sachin Patwardhan for providing a stimulating academic

environment.

I thank Dr. Rajan for his dedication and my friends Arun and Krishnaji for their patience during my weekend violin lessons. I thank my friends, Arun, Arvind, Bhushan, Errol, Govind, Praveen, Raghu, Ramesh, Sanket, Shakir, Sujit, Suresh, Venkat, Vikas and Vinay for the wonderful moments we shared in Edmonton.

I am indebted to my parents, my sister and brother-in-law for their love and support without which this work would not have been possible. I am grateful to my parents and family members for their faith and trust in me. I am grateful to my parents and Mathaji for revealing spiritual dimensions of the scientific aspect of God.

Contents

1 **Introduction** **1**
- 1.1 What is Fault Detection and Diagnosis (FDD)? 1
  - 1.1.1 Basic tasks of an FDD routine 1
- 1.2 Need for FDD 3
  - 1.2.1 Consequences of sensor faults in systems operating under closed-loop conditions 3
  - 1.2.2 Illustration using an exothermic CSTR process 3
- 1.3 Basic principles of FDD 6
  - 1.3.1 Model selection 7
  - 1.3.2 Residual generation 8
  - 1.3.3 Decision making 8
- 1.4 Diagnostic task 8
  - 1.4.1 Process Monitoring 9
  - 1.4.2 Input-output faults 9
  - 1.4.3 Additive Process faults 10
  - 1.4.4 Multiplicative Process faults 10
- 1.5 Organization of this thesis 10
  - 1.5.1 Process monitoring and FDI using steady-state models 11
  - 1.5.2 Input-output FDI using discrete-time models 12
  - 1.5.3 Model identification for processes with irregularly sampled outputs 12
  - 1.5.4 Additive Process FDI using continuous-time models 13

2 **Steady-state model identification and fault diagnosis** **14**
- 2.1 Overview 14
- 2.2 Steady-state model identification 15
  - 2.2.1 Multiple Linear Regression (MLR) 17
  - 2.2.2 Errors-in-variables (EIV) Modeling 17

2.2.3 Principal Components Analysis (PCA) 18
2.2.4 Iterative Principal Components Analysis (IPCA) 19
2.2.5 Partial Least Squares (PLS) 20
2.3 Bitumen Recovery Prediction 22
2.3.1 Process description 22
2.3.2 Challenges in soft-sensor development 24
2.3.3 Data Acquisition and Import 24
2.3.4 Data Quality Assessment 25
2.3.5 Online Results 36
2.4 Steady-state Fault Detection and Isolation (FDI) 42
2.4.1 FDI using PCA and contribution plots 42
2.5 Turbine Failure Diagnosis 44
2.5.1 Immediate reasons for the trip 44
2.5.2 Diagnosis of Bearing Fatigue 45
2.6 Isolation enhancements using structured residuals 49
2.6.1 Maximized sensitivity enhancements 50
2.7 FDI in a Quadruple tank process 53
2.8 Conclusions 57

3 Discrete-time dynamic model identification and fault diagnosis 59
3.1 Overview 59
3.2 I/O faults in open-loop and closed-loop systems 60
3.2.1 Open loop - actuator faults - Trend plots 62
3.2.2 Open loop - sensor faults - Trend plots 64
3.2.3 Closed loop case - actuator faults - Trend plots 64
3.2.4 Closed loop case - sensor faults: Trend plots 66
3.3 Diagnosis of I/O faults using the SRV approach 68
3.4 Closed-loop FDD using the SRV approach 70
3.4.1 Expermimental evaluation on a pilot-scale Continuous-flow Stirred Tank Heater (CSTH) process 72
3.5 I/O fault diagnosis in a Quadruple tank process 76
3.5.1 Fault diagnosis using PCA 76
3.5.2 Fault diagnosis using the SRV approach 80
3.6 Conclusions 84

4 Identification of processes with irregular output sampling 87

4.1 Overview 87
4.2 Introduction 88
4.3 Problem description 92
    4.3.1 Model Structure and Assumptions 92
    4.3.2 Identification problem 93
    4.3.3 Estimation problem 93
4.4 EM-based state space model identification 93
    4.4.1 Maximization of the likelihood function 94
    4.4.2 Kalman filter and smoother expressions 94
    4.4.3 Modified Kalman filter for missing data 97
    4.4.4 EM algorithm 97
4.5 Identification of the initial model 102
    4.5.1 Expressions for new model parameters in the M-step 103
4.6 Handling missing observations in the EM-based identification . . . 104
4.7 Illustrative applications 105
    4.7.1 Example: Simplified version of EM procedure 105
    4.7.2 Simulated Case-study: 3^{rd} order underdamped system 107
    4.7.3 Laboratory Case-study: Continuous-flow Stirred Tank Heater (CSTH) process 110
    4.7.4 Industrial Case-study: Bleaching unit in a BCTMP mill 112
4.8 Conclusions 115

5 Gray-box model identification in a pulp mill 117
5.1 Overview 117
5.2 Introduction 118
5.3 Process Description 121
5.4 Identification of processes with irregularly sampled outputs 126
    5.4.1 What is wrong with arbitrary interpolation? 127
    5.4.2 Methods for handling missing data 128
    5.4.3 The EM algorithm 130
    5.4.4 How irregular can the sampling be? 130
5.5 Identification of Delay-dominant recycle processes 133
    5.5.1 Estimation of consistent forward and recycle models 135
5.6 Other practical considerations 139
    5.6.1 Structure of the nonlinearity 139

5.6.2 Identification from routine operating data 139

5.7 Results 140

5.7.1 Results using black-box identification 140

5.7.2 Results using linear interpolation 141

5.7.3 From black-box models to gray-box model model structures 143

5.7.4 Results using constrained FIR and EM algorithm 144

5.8 Conclusions 150

6 Diagnosis of additive process faults using continuous-time methods 153

6.1 Overview 153

6.2 Introduction 154

6.3 Motivation and Problem Formulation 156

6.3.1 Motivation 156

6.3.2 Problem Formulation 158

6.4 Overview of the Numerical Integrators 161

6.5 Identification of the PRM 162

6.5.1 Description of the Identification problem 162

6.5.2 Consistent Estimation of $\mathbf{\Gamma}_s$ 164

6.5.3 Calculation of $\mathbf{W}_0$ 166

6.5.4 Identification of $\mathbf{W}_0\mathbf{H}_s$. 166

6.5.5 Consistent Estimation of $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ 167

6.6 Design of the SRMs for Fault Isolation 168

6.7 Numerical Results 170

6.8 Conclusions 178

7 Conclusions 179

7.1 Summary 179

7.2 Contributions of this thesis 181

7.3 Recommendations for future work 182

A Comparison of structured residual fault isolation techniques 184

A.1 FDI performance tables 184

B Sample Matlab programs for the EM algorithm 197

B.1 Matlab code for EM algorithm 197

B.2 Matlab code for Kalman Smoother 201

B.3 Matlab code for Initial FIR model 205

Bibliography **207**

List of Tables

1.1 *Exothermic CSTR parameters* 4
1.2 *CSTR Equilibrium points* 4

2.1 Process variables used in soft-sensor 37
2.2 *Singular values obtained using the IPCA procedure* 38
2.3 *Performance indices for the soft-sensors* 38
2.4 *Incidence matrix to characterize the isolation logic* 51
2.5 *Different FDI settings for montecarlo experiments* 55
2.6 *Incidence matrix to characterize the isolation logic* 57

3.1 *Parameter values and chosen operating point of the Quadruple-tank process* 62
3.2 *Open loop Actuator Bias diagnosis using PCA* 76
3.3 *Open loop Sensor Bias diagnosis using PCA* 76
3.4 *Closed loop Actuator Bias diagnosis using PCA* 77
3.5 *Closed loop Sensor Bias diagnosis using PCA* 77

6.1 *Incidence matrix to characterize the isolation logic* 170
6.2 *Parameter values and chosen operating point of the Quadruple-tank process* 170
6.3 *Correspondences between the PRM and the fault gain matrix in the CT domain and their counterparts in the DT domain* 176

A.1 *Results of FDI using scalar structured residual approach with multiple testing - Part1* 185
A.2 *Results of FDI using scalar structured residual approach with multiple testing - Part2* 186

A.3 *Results of FDI using scalar structured residual approach with maximized sensitivity and multiple testing - Part1* 187
A.4 *Results of FDI using scalar structured residual approach with maximized sensitivity and multiple testing - Part2* 188
A.5 *Results of FDI using structured residual vector approach with maximized sensitivity and multiple testing - Part1* 189
A.6 *Results of FDI using structured residual vector approach with maximized sensitivity and multiple testing - Part2* 190
A.7 *Results of FDI using scalar structured residual approach with single testing - Part1* 191
A.8 *Results of FDI using scalar structured residual approach with single testing - Part2* 192
A.9 *Results of FDI using scalar structured residual approach with maximized sensitivity and single testing - Part1* 193
A.10 *Results of FDI using scalar structured residual approach with maximized sensitivity and single testing - Part2* 194
A.11 *Results of FDI using structured residual vector approach with maximized sensitivity and single testing - Part1* 195
A.12 *Results of FDI using structured residual vector approach with maximized sensitivity and single testing - Part2* 196

List of Figures

1.1 Schematic representation of a CSTR with an exothermic first order chemical reaction . . . 3
1.2 Nature of fault: Sensor drift . . . 5
1.3 Difference between observation and actual temperature . . . 5
1.4 Trend plots of the sensor measurements . . . 6
1.5 A typical FDD loop . . . 7
1.6 Techniques for Fault detection and diagnosis . . . 7
1.7 Types of faults affecting linear dynamic systems . . . 9

2.1 Process Flow-sheet for Separation Cell . . . 23
2.2 Data-based view of an industrial process . . . 24
2.3 Swinging Door algorithm for data compression . . . 27
2.4 Example of a flow tag getting censored . . . 28
2.5 Example of severe quantization in a conductivity tag . . . 28
2.6 Sample Consolidation mechanism . . . 30
2.7 MSE as a function of delay when $\omega = 1$ . . . 33
2.8 MSE as a function of delay for various ω values . . . 34
2.9 MSE as a function of delay when $\omega = 0.2$ . . . 34
2.10 Tailings distribution shows significant deviation from normality . . . 35
2.11 Linear model does not predict spikes adequately . . . 36
2.12 Predictions improve after logarithmic transformation . . . 36
2.13 Plot of the singular values obtained through IPCA procedure . . . 39
2.14 Plot of the cumulative X and Y variances captured by successive PLS dimensions . . . 39
2.15 Predictions of Tailings Bitumen using PLS and IPCA techniques . . . 40
2.16 Predictions of Flotation Tailings Bitumen using PLS and IPCA techniques . 40
2.17 Predictions of Froth Bitumen using PLS Regression . . . 41

2.18 Online prediction of Bitumen Recovery 41
2.19 Fault detection using the Squared Prediction Error 45
2.20 Contribution plot at sample 10551 45
2.21 Percentage contribution of oil pressure to SPE 46
2.22 Absolute contribution of oil pressure to SPE 46
2.23 Trend plot of oil pressure 46
2.24 Deviation in SPE - Initial stages 47
2.25 Definition of Eccentricity 47
2.26 Turbine Bearings 48
2.27 Trend plots of Bearing temperatures, Eccentricity and Steam temperature . 49
2.28 Flow network with 8 measurements and 3 constraints 51
2.29 Fault directions orthogonal to fault in F_8 53
2.30 Schematic of the four water tank system 54
2.31 Steady-state flow network for quadruple tank process 54

3.1 Schematic of the four water tank system 61
3.2 Schematic illustrating the introduction of actuator biases 61
3.3 Schematic illustrating the introduction of sensor biases 61
3.4 Actuator Biases: Differences between the expected inputs and true inputs . 63
3.5 Actuator Biases: Trend plots of the observed variables 63
3.6 Sensor Biases: Differences between the actual state of process and observed state of the process 64
3.7 Sensor Biases: Trend plots of the observed variables 65
3.8 Closed-loop Actuator Biases: Differences between the expected inputs and true inputs 65
3.9 Closed-loop Actuator Biases: Trend plots of the observed variables 66
3.10 Closed-loop Sensor Biases: Differences between the observed and true outputs 67
3.11 Closed-loop Sensor Biases: Trend plots of the observed variables 67
3.12 Continuous-flow Stirred Tank Heater 71
3.13 Temperature Sensor Bias under closed-loop conditions 71
3.14 Water level: Comparison of model predictions with measured output 72
3.15 Water Temperature: Comparison of model predictions with measured output 72
3.16 Temperature Sensor Faults introduced under closed-loop conditions 73

3.17 SRV-based FDI indices for -0.1mA sensor bias under closed-loop conditions 73
3.18 SRV-based FDI indices for -0.2mA sensor bias under closed-loop conditions 74
3.19 SRV-based FDI indices for -0.3mA sensor bias under closed-loop conditions 74
3.20 SRV-based FDI indices for -0.5mA sensor bias under closed-loop conditions 75
3.21 SRV-based FDI indices for ramp-type fault under closed-loop conditions . . 75
3.22 SPE plot for open-loop 1σ actuator bias in u_1 77
3.23 T^2 plot for open-loop 1σ actuator bias in u_1 78
3.24 2-D score plot for 3σ open loop actuator bias in u_1 78
3.25 SPE plot for 6σ open loop sensor bias in y_1 79
3.26 Contribution plot for 6σ open loop sensor bias in y_1 79
3.27 SPE plot for 6σ closed loop actuator bias in u_1 80
3.28 T^2 plot for 6σ closed loop actuator bias in u_1 80
3.29 SPE plot for 3σ closed loop sensor bias in y_1 81
3.30 T^2 plot for 3σ closed loop sensor bias in y_1 81
3.31 Detection of open-loop actuator faults using the SRV approach 82
3.32 Isolation of 1σ actuator fault in u_2 82
3.33 Detection of open-loop sensor faults using the SRV approach 83
3.34 Isolation of 3σ sensor fault in y_2 83
3.35 Detection of closed-loop actuator faults using the SRV approach 84
3.36 Isolation of 1σ actuator fault in u_1 84
3.37 Detection of closed-loop sensor faults using the SRV approach 85
3.38 Isolation of 3σ sensor fault in y_1 85

4.1 Plot of θ_1 and θ_2 as a function of number of iterations 107
4.2 Plot of the average prediction error as a function of number of iterations . . 107
4.3 Singular value plot for order selection 108
4.4 Monitoring convergence using the Negative log likelihood function 109
4.5 Comparison of step responses of the underdamped system 109
4.6 Comparison of Kalman filter predictions with true output in the underdamped system 110
4.7 Schematic of a pilot-scale CSTH process 111
4.8 Comparison of PEM and EM step responses 111

4.9 Comparison of Kalman filter predictions with true output 112
4.10 Simplified flow sheet of Mechanical Pulp mill 113
4.11 Distribution of the output sampling intervals 113
4.12 Predictions using linear interpolation look good 114
4.13 Step responses using interpolated data are not satisfactory 115
4.14 Brightness predictions using EM-based strategy 116
4.15 Step responses using EM-based strategy 116

5.1 Line 1 and Line 2 Brightness measurements 123
5.2 Flow sheet of BCTMP bleaching unit 124
5.3 Simplified flow sheet of bleaching operation in a BCTMP mill 125
5.4 Distribution of the sampling intervals for the output variables 125
5.5 Various types of interpolation 128
5.6 Block diagram of Gain-delay-recycle system 131
5.7 Block diagram of a typical recycle system 134
5.8 A typical staircase like step response of a recycle system 135
5.9 Predictions from N4SID look good 141
5.10 N4SID step responses are not satisfactory 141
5.11 Predictions using linear interpolation look good 142
5.12 Step responses using interpolated data are not satisfactory 142
5.13 Deterioration of predictions for model based on linearly interpolated data . 143
5.14 Choosing the best model using minimum RMSE 146
5.15 Step response of Brightness for unit change in P1 Peroxide 147
5.16 Block diagram showing implementation of predictors for Line 1 148
5.17 Snapshot of Line 1 predictions 148
5.18 Line 1 Brightness Predictions 149
5.19 Line 2 Brightness Predictions 149
5.20 Step responses of FIR-EM Brightness model are satisfactory 150
5.21 Line 1 Tensile Predictions 150
5.22 Line 2 Tensile Predictions 151
5.23 Line 1 Tensile Predictions (Jan - May 2003) 151
5.24 Line 2 Tensile Predictions (Jan - May 2003) 152

6.1 Schematic of the four water tank system 157
6.2 Selection of T_s and l for continuous time system 171
6.3 CT residual model-based detection and isolation of a leak in Tank 1 172

6.4 CT residual model-based detection and isolation of two simultaneous leaks in Tanks 3 and 4 173
6.5 CT residual model-based detection and isolation of three simultaneous leaks in Tanks 2, 3 and 4 173
6.6 CT residual model-based detection and isolation of simultaneous leaks in all 4 Tanks 174
6.7 CT residual model-based detection and isolation of two simultaneous incipient leaks in tanks 3 and 4. 174
6.8 The CT residual model-based FDI scheme successfully detects and isolates a time-varying leak in Tank 3 176
6.9 The DT residual model-based FDI scheme fails to detect and isolate a time-varying leak in Tank 3 177
6.10 The DT residual models can detect and isolate a leak with a ZOH in Tank 3 , with relatively poor performance 177

Abbreviations

ARX	Auto Regressive models with eXogenous inputs
BCTMP	Bleached-Chemi Thermo-Mechanical Pulp
CC	Correlation Coefficient
CSTH	Continuous-flow Stirred Tank Heater
CSTR	Continuous-flow Stirred Tank Reactor
CT	Continuous-Time
CVA	Canonical Variate Analysis
DCS	Distributed Control System
DT	Discrete-Time
EIV	Errors-In-Variables
EM	Expectation Maximization
EWMA	Exponentially Weighted Moving Average
FDD	Fault Detection and Diagnosis
FDI	Fault Detection and Isolation
FDRI	Fault Diagnosis Relevant Identification
FIR	Finite Impulse Response
FTCS	Fault-Tolerant Control System
IPCA	Iterative Principal Components Analysis

LTI	Linear time invariant
MCMC	Markov-Chain Monte-Carlo
MI	Multiple Imputation
MLE	Maximum Likelihood Estimate
MLR	Multiple Linear Regression
MSE	Mean Squared Error
NLL	Negative Log Likelihood
OLS	Ordinary Least Squares
PC	Principal Component
PCA	Principal Components Analysis
PCR	Principal Components Regression
PEM	Prediction Error Methods
PFDI	Process Fault Detection and Isolation
PLS	Partial Least Squares
PLSR	Partial Least Squares Regression
PRM	Primary Residual Model
PRV	Primary Residual Vector
RMSE	Root Mean Squared Error
SIM	Subspace Identification Methods
SISO	Single-input Single-output
SPE	Squared Prediction Error
SRAMS	Structured Residual Approach with Maximized Sensitivity
SRM	Structured Residual Model

SRV	Structured Residual Vector
SVD	Singular Value Decomposition
SWR	Squared Weighted Residual
T^2	Hotelling's T^2 statistic
ZOH	Zero-Order-Hold

1

Introduction

1.1 What is Fault Detection and Diagnosis (FDD)?

A common problem within the process industry is that many control loops do not operate in a satisfactory manner. Though the controllers are designed to be robust to model uncertainty and disturbance rejection, they rarely work in a "fault-tolerant" manner. A simplistic and very general definition of a fault is that, it is an unacceptable deviation of properties or variables of a system. Typical problems are for instance, oscillations, sensor offsets and drifts, valve stiction, leaks, parameter changes *etc*. Since a typical plant consists of hundreds or thousands of control loops, which interact in a rather complex way, it is very hard to detect and diagnose such problems.

1.1.1 Basic tasks of an FDD routine

The main tasks of an FDD routine can be classified into two categories, detection and diagnosis. The diagnosis task can be further subdivided into, isolation, identification, and reconfiguration. When the FDD routine is used as an advisory system, its task is restricted to fault detection and isolation. When it is a part of an Abnormal Situations Management (ASM) system, it needs to have further capabilities. In this connection two possibilities exist. One can think of the FDD routine as a part of a Fault-Tolerant Control System (FTCS). In this case, the system continues to operate even in the presence of a fault. For example, this is possible in the presence of sensor and actuator faults. FTCS in the presence of sensor faults is more common. If it is possible to reconstruct the value of the sensor, or more importantly, infer the underlying state, it is possible to continue running the process even in the presence of the sensor fault. Similarly, if the system possesses enough redundant actuators, for example, as in an aircraft wing, it is possible to continue running the process even though some actuators have failed. The other possibility is to consider the FDD routine as a part of an alarm management system. Under the presence of certain faults, the FDD routine can trigger a *safe* shutdown before the occurrence of a catastrophic event. Different combinations of the following components of FDD help achieve the above mentioned capabilities:

Detection

The task in fault detection is to indicate whether a fault exists in the system under observation. This is a *Yes/No* question. It depends on the question of what constitutes a fault. While it has deep philosophical implications, a pragmatic viewpoint is to leave this decision to the user. However, it is important to ensure that the ignored faults do not develop into abnormal events which are of concern to the user.

Isolation

The task in fault isolation is to find the location of the fault. This is one of the most difficult tasks in the FDD process. It is made easier if a good model of the system is available. In this connection, a good model can be considered as one that has enough information, or level of detail, for the diagnostic task at hand.

Identification

The task in fault identification is to estimate the size of the fault. It is used mainly for FTCS. Fault identification is accomplished by state estimation using physical and analytical redundancy present in the system. The size of the fault is determined by comparing the values of the states computed using a number of different ways.

Reconfiguration

The presence of a fault generally makes the currently used control configuration sub-optimal or infeasible. Once the size of the fault has been estimated, and the decision to continue running the process has been made, a switch is made to a feasible configuration using a pre-determined logic. This is known as reconfiguration.

1.2 Need for FDD

Before looking at different kinds of faults and methods used to diagnose the presence of these faults, it is important to consider whether it is necessary to really have a fault diagnosis routine. We provide the justification for such a routine using a simple simulated case-study.

1.2.1 Consequences of sensor faults in systems operating under closed-loop conditions

When sensor faults develop under closed loop conditions, the consequences to the health of the process can be significant. We illustrate this using a simulation of this condition in a continuous-flow stirred tank reactor in which an exothermic reaction takes place.

1.2.2 Illustration using an exothermic CSTR process

Consider the operation of a Continuous-flow Stirred Tank Reactor (CSTR) in which an exothermic first-order chemical reaction with Arrhenius temperature dependence takes place (Marlin 1995). A schematic representation of the process is shown in Fig. 1.1.

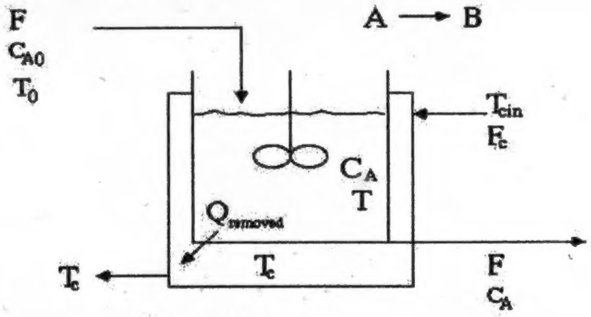


Figure 1.1: Schematic representation of a CSTR with an exothermic first order chemical reaction

The objective is to control the temperature and hence the concentration of the reactant in the CSTR. This is achieved by manipulating the flow rate of the coolant. This system is described by the following mass and energy balance relations (Marlin 1995):

$$\begin{aligned} V\frac{dC_A}{dt} &= F(C_{A0} - C_A) - Vk_0e^{-\frac{E}{RT}}C_A \\ V\rho C_v\frac{dT}{dt} &= \rho C_v F(T_0 - T) - \frac{aF_c^{b+1}}{F_c + \frac{aF_c^b}{2\rho_c C_{pc}}}(T - T_{cin}) + (-\Delta H_{rxn})Vk_0e^{-\frac{E}{RT}}C_A \end{aligned} \tag{1.1}$$

This system has multiple equilibrium points. These are obtained by setting $\frac{d\mathbf{x}}{dt} = 0$, where, $\mathbf{x} = [C_A \; T]^T$. Upon linearizing the equations around these points, two of the three points lead to linear models which have poles in the left-half s-plane while the third leads to a linear model which has a pole in the right-half s-plane. The physical parameters of the process is given in the following table:

Table 1.1: *Exothermic CSTR parameters*

F - Inlet flow rate	$1m^3/min$
V - Volume of Reactor	$1m^3$
C_{A0} - Inlet Concentration	$2.0kmole/m^3$
C_p - Specific Heat capacity of reactant	$1cal/g^oC$
ρ - Density of reactant	$10^6g/m^3$
k_0 - Reaction Constant	$10^{10}min^{-1}$
E/R - Arrhenius factor	$8330.1K^{-1}$
ΔH_{rxn} - Heat of reaction	$130 \times 10^6 cal/kmole$
$(F_c)_s$ - Steady state Coolant flow rate	$15m^3/min$
C_{pc} - Specific Heat capacity of coolant	$1cal/(gK)$
ρ_c - Density of coolant	$10^6g/m^3$
b - Reaction related parameter	0.5
T_0 - Inlet temperature	$343K$
T_{cin} - Coolant Inlet temperature	$310K$
a - Reaction related parameter	$0.516 \times 10^6(cal/min/K)/(m^3/min)$

The following additional parameters correspond to the three equilibrium points:

Table 1.2: *CSTR Equilibrium points*

Equilibrium point $C_A(kmole/m^3)$	1.79	1.37	0.16
Equilibrium point $T(K)$	330.9	349.88	404.7
Nature of equilibrium point	Stable	**Unstable**	Stable

It is possible to operate this CSTR at the unstable point with the help of feedback control. Thus if a feedback PI controller is implemented to maintain temperature at 350 K by adjusting the coolant flow rate, the system reaches the intermediate state.

We now show that operating this process at the *open-loop* unstable point in the presence of a sensor drift, leads to undesirable behavior. Towards this end, we have designed the PI controller $G_c(s) = 1.5 + \frac{0.5}{s}$ to stabilize the system. We assume that the temperature sensor has an additive measurement noise characterized by a zero-mean gaussian process with standard-deviation $0.71K$. A drift, as shown in Fig. 1.2 is introduced in the sensor measurement. We show that this sensor fault results in undesirable oscillations.

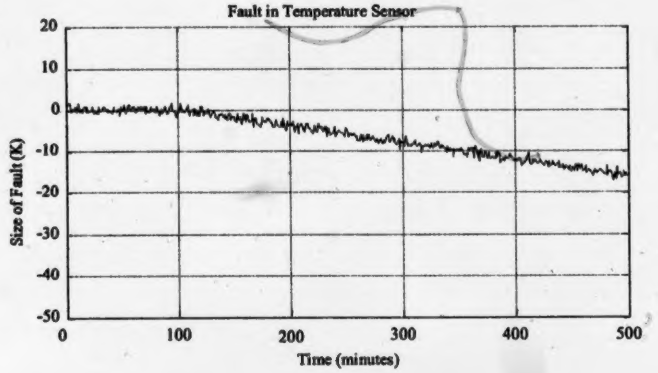


Figure 1.2: Nature of fault: Sensor drift

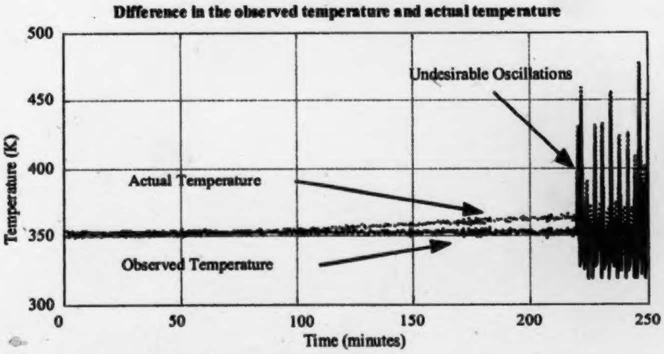


Figure 1.3: Difference between observation and actual temperature

There can be a large difference between the observed temperature and the actual temperature. This is illustrated in Fig. 1.3. While the sensor fault sets in, there is apparently no change in the sensor reading seen by the operator.

The drift is introduced in the sensor at $t = 100$ minutes. The trend plots of the sensor measurements are shown in Fig. 1.4. The controller changes the coolant flow rate under the assumption that the drift is due to a disturbance. As a result, the reaction rate and hence the concentration starts deviating from the desired value, while the sensor reading inclusive of the drift is held constant. As a result, the sensor fault cannot be detected from the observed temperature trend plot. The movement in the concentration and the manipulated variable could be easily mistaken as the effect of a disturbance in the system. The fact that something is wrong in the system can be found only when undesirable oscillations set in

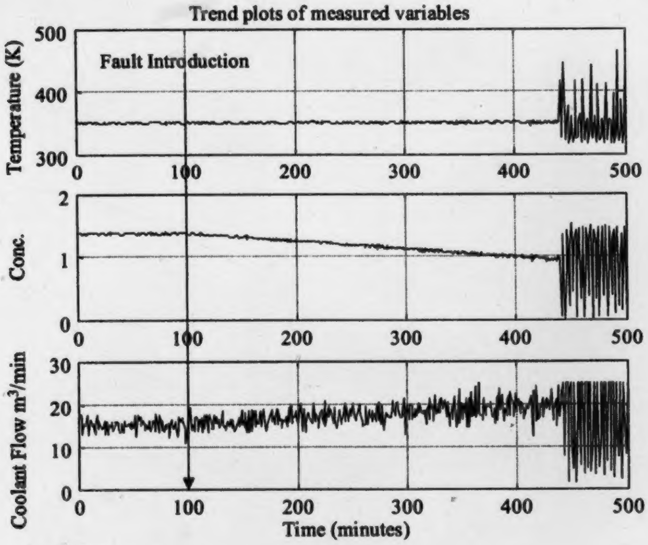


Figure 1.4: Trend plots of the sensor measurements

and after this, it is very difficult to trace it back to the error in the temperature sensor.

While this example might appear extreme, there are other cases where the presence of faults may degrade the performance of the system. In the latter part of this thesis, we show that such faults result in a degradation in the performance of systems operated under open or closed-loop conditions. We illustrate the differences in observed and actual process inputs and outputs in a simulated Quadruple-tank case-study under open-loop and closed-loop conditions in Sec. 3.2. In light of these safety concerns and performance degradation, the presence of an FDD system has important benefits.

1.3 Basic principles of FDD

One of the shortcomings of research in the area of Fault Detection and Diagnosis (FDD) is that the methods developed are rarely applicable to process plants. A significantly large amount of research in FDD assumes the availability of an accurate mathematical model of the system under observation. The role of an accurate model for Fault Detection and Diagnosis is illustrated in Fig. 1.5. Some of the existing techniques are shown in Fig. 1.6. It is evident from Fig. 1.5 that any fault detection and diagnosis routine consists of three parts, model selection or identification, residual generation and processing, and decision

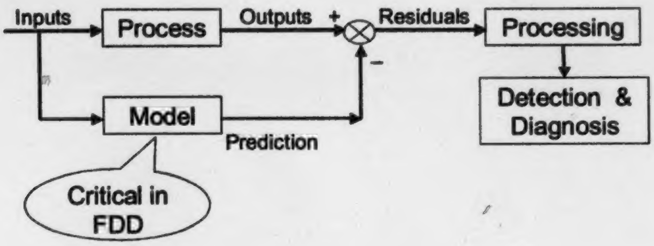

Inputs
Process
Outputs
+
Residuals
Processing
−
Model
Prediction
Detection & Diagnosis
Critical in FDD

Figure 1.5: A typical FDD loop

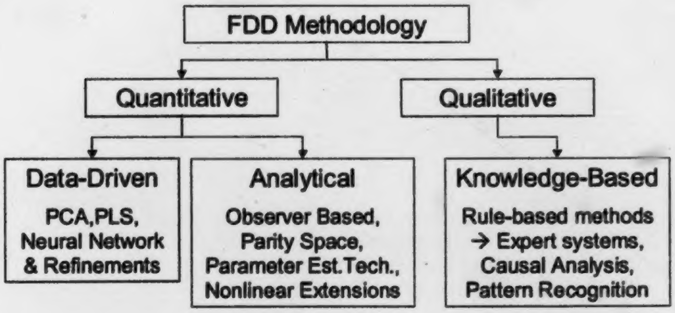

FDD Methodology
Quantitative
Qualitative
Data-Driven
PCA,PLS, Neural Network & Refinements
Analytical
Observer Based, Parity Space, Parameter Est.Tech., Nonlinear Extensions
Knowledge-Based
Rule-based methods → Expert systems, Causal Analysis, Pattern Recognition

Figure 1.6: Techniques for Fault detection and diagnosis

making. We can summarize these tasks as follows:

1.3.1 Model selection

Results obtained from FDD routines depend on the type of models used. FDD routines and the models they are built upon can be categorized as qualitative and quantitative. In addition, the amount of detail present in the model affects the performance of the FDD routine. For example, first-principles models that contain enough detail can be used to give more accurate and detailed diagnosis than simpler data-based models.

However, the nature of the model should depend on the diagnostic task at hand. For example, for simple process monitoring applications which mainly concentrate on fault detection, using a data-based steady-state model may be enough. However, if one needs to isolate complex process faults, it might be necessary to use a detailed first-principles model. In this thesis we concentrate on quantitative data-based models.

1.3.2 Residual generation

Once a model is obtained, an important component of the FDD routine is residual generation. Information from the process, in terms of sensor readings and manipulated input values, is used along with the model to generate residuals which are to be used for fault detection and isolation. An important classification of the quantitative FDD methods is based on the way in which these methods process the residuals. For example, simple steady-state data-based methods like PCA use correlation analysis to produce contribution plots for fault diagnosis (Russell *et al.* 2000, Kresta *et al.* 1991). On the other hand, some quantitative model-based techniques process the residuals to obtain enhanced residuals with structural or directional properties (Gertler 1998). Alternatively special observers are constructed for processing the residuals to achieve specific detection and isolation objectives (Frank 1990). In addition, the statistical local approach (Basseville and Nikiforov 1993) provides a framework for fault detection in terms of detecting small changes in parameters and transforms these into a problem of monitoring the mean of a Gaussian vector.

1.3.3 Decision making

This task involves making a decision about the presence of a fault and its location, using the information present in the residuals. In most quantitative model-based FDD techniques it involves comparison of fault detection and isolation indices computed using the residuals, with confidence limits calculated using a stochastic characterization of the residual space, based on data collected under normal, fault-free conditions.

In the structured residual approaches (Gertler and Singer 1990), which are the main techniques used in this thesis, one can consider two different approaches for decision making. For the single faults hypothesis, we use the term *multiple testing* to denote the decision made by comparing the complete set of fault isolation indices based on the structured residual with their confidence limits. On the other hand, we use the term *single testing* when the smallest of the fault isolation indices rather then the entire set of fault isolation indices is used to make a decision.

1.4 Diagnostic task

Some of the types of faults which affect physical systems are illustrated in Fig. 1.7. Generally, it is easier to detect and diagnose input-output (I/O) faults like sensor and

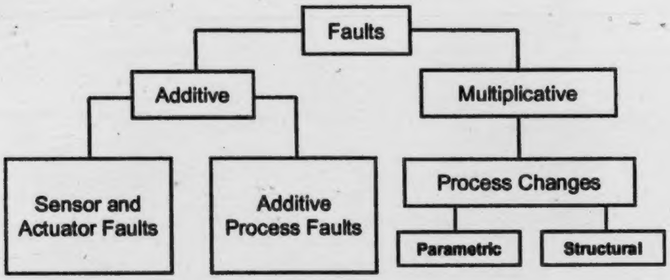


Figure 1.7: Types of faults affecting linear dynamic systems

actuator faults than to detect and diagnose process faults. Among process faults, it is easier to detect and isolate additive process faults than multiplicative process faults which arise from parametric changes. The information required for effective fault detection and diagnosis is related to the diagnostic task on hand. We can consider, for example, four classes of diagnostic tasks, process monitoring, sensor and actuator FDI, additive process fault detection and diagnosis and multiplicative process fault diagnosis.

1.4.1 Process Monitoring

The term *process monitoring* is generally used to refer to the task of detecting faults or changes in the operating condition in a process. In addition, some indication is provided as to the cause of the abnormality, if it exists. Using this information, the operator can decide whether any action is to be taken to bring back the process to its normal operating regime, or whether the fault requires immediate attention. Process monitoring is usually applied using simple steady-state data-based models which are constructed using routine operating data. Some of the popular techniques for process monitoring include those based on multivariate statistical techniques like PCA or PLS (Russell *et al.* 2000, Kresta *et al.* 1991).

1.4.2 Input-output faults

It is generally assumed that sensor and actuator faults affect the rest of the system in an additive way for linear systems. They can be diagnosed using discrete-time input/output (I/O) models of processes since they affect processes at the input or output ports. Such I/O models can be identified using standard techniques. Following the identification of these I/O models, sensor and actuator faults can be detected and diagnosed under open-loop and closed-loop conditions, using model based techniques such as, parity-space, observer-based techniques or their extensions.

1.4.3 Additive Process faults

The dynamics of most physical systems can be accurately represented by continuous-time (CT) state-space models. Representation of CT systems by discrete-time (DT) models is only an approximation of their dynamics. Since it is easier to identify DT models than CT models and as in many cases a CT system's dynamics can be represented well by a DT model, DT models have been widely used with success. However, in some cases, e.g. in the isolation of process faults, one has to use CT models. For example, while detection and isolation of sensor and actuator faults in a CT system can be performed using its discrete-time (DT) state space model, process fault detection and isolation requires the knowledge of the fault gain matrix, which links the process faults to the state variables of the system. For the CT state space model of the system, the fault gain matrix is always available no matter how the faults are varying with time, while in the DT state space model of the system, the fault gain matrix is available only when the faults follow some known functions of time, e.g. if they are piece-wise constant. In the case of additive process faults, it is possible to identify fault-diagnosis relevant models for detecting and isolating these faults, using subspace identification techniques (Li *et al.* 2003).

1.4.4 Multiplicative Process faults

Multiplicative process faults arise due to changes in the physical parameters of a system (Basseville and Nikiforov 1993). Isolation of such faults using DT models of the system is not trivial because the change in a physical parameter associated with a system can result in changes in several parameters in the DT model. In contrast there usually exists a one-to-one correspondence between the CT model parameters and the physical parameters of the system (Isermann 1984). Hence it is necessary to use CT models for such faults.

1.5 Organization of this thesis

As discussed earlier, FDD techniques differ based on the type of model used. The fact that good models are not readily available for many physical processes makes a majority of the approaches unsuitable for application to solve problems faced by process industries. The focus of this thesis is on data-based techniques for fault diagnosis relevant model identification. Special emphasis is placed on identification of fault diagnosis-relevant models of the system under consideration. Fault diagnosis relevant model identification attempts to build models of the system so that the models developed are compatible with the detection and diagnosis of faults or events under investigation. Often the diagnosis results achievable by the FDD routine are limited by the quality of the instrumentation available in the system, the data quality and hence the quality of the model obtained from the data. This affects the set of diagnosable faults in the system. As discussed earlier, simple process monitoring is possible by relatively simple steady-state, data-driven, models. However, to diagnose complex process faults, more detailed first-principles models may be required. Considering these differences in the diagnostic ability of the methods, we study the problems which arise from the application of these methods to solve problems of process monitoring and FDI in a variety of industrial, laboratory and simulation case-studies and propose modifications and new approaches for solving these problems. The thesis is organized into the following parts:

- Process monitoring and FDI using steady-state models.
- Input-output FDI using discrete-time models.
- Model identification for processes with irregularly sampled outputs.
- Additive Process FDI using continuous-time models.

1.5.1 Process monitoring and FDI using steady-state models

When the process remains close to a particular steady-state, static models developed based on multivariate statistical data-compression techniques using routine operating data can be used for process monitoring (Russell *et al.* 2000, Kresta *et al.* 1991). However the fault isolation techniques used with these methods are correlation-based and have to be supplemented with process knowledge supplied by the user. These methods have gained wide acceptance because of their simplicity, data-crunching ability, and the availability of process knowledge from operator experience. It is possible to extend the isolation capabilities of these methods using the concepts of structured residuals (Gertler *et al.* 1999, Qin and Li 1999, Li and Shah 2002) which utilize information available in the model to project the residual vector into special subspaces. This results in improved fault diagnosis. We illustrate these ideas in this thesis using the following case-studies:

- Failure diagnosis in a thermal power plant using Principal Components Analysis (PCA) (Ch. 2, Sec. 2.5).
- Inferential Sensor Development for the prediction and monitoring of Bitumen Recovery in a Separation Cell of an Oil-Sands Extraction plant using the Partial Least

Squares (PLS) and Iterative PCA (IPCA) techniques (Ch. 2, Sec. 2.3). *Due to the proprietary nature of these processes, the documentation for the process description is not as complete as one would like to have.*

- Structured Residual approaches for FDI in a quadruple tank process under steady-state conditions (Ch. 2, Sec. 2.7).

1.5.2 Input-output FDI using discrete-time models

When faults develop in the actuators or sensors associated with a process, there is a discrepancy between the actual input to the process or output from the process and the input given to the process or information received from the process. Models of the process, identified using techniques such as Prediction Error Methods (PEM) (Ljung 1999) or Subspace Identification Methods (SIM) (Viberg 1995, Overschee and DeMoor 1996) can be used with model-based FDI techniques (Gertler 1998, Frank 1990) for the detection and isolation of sensor and actuator faults under open-loop and closed-loop conditions. We illustrate these ideas in this thesis using the following case-studies:

- Sensor Validation in closed-loop systems: Experimental studies on a pilot-scale Continuous-flow Stirred Tank Heater (Ch. 3, Sec. 3.4).
- Comparison of the PCA and SRV techniques for Sensor and Actuator FDI: Simulation studies on a Quadruple-tank process (Ch. 3, Sec. 3.5).

1.5.3 Model identification for processes with irregularly sampled outputs

In many chemical processes, variables which indicate product quality are infrequently and irregularly sampled. However, the application of traditional identification techniques, like Prediction Error Methods (PEM) (Ljung 1999) and Subspace Identification (Overschee and DeMoor 1996) to processes with irregular output sampling is a non-trivial task. When the quality variables are irregularly sampled, Maximum Likelihood Estimation of the model parameters can be performed using the Expectation Maximization (EM) approach (Dempster *et al.* 1977, Shumway and Stoffer 1982). This identification procedure also yields a Kalman filter-based prediction-correction mechanism which can be used for optimal prediction of the quality variables. We illustrate optimal model identification and output prediction for processes with irregularly sampled outputs in this thesis through the following case-studies:

- Simulated Case-study: 3^{rd} order underdamped state-space model (Ch. 4).
- Laboratory Case-study: Continuous-flow Stirred Tank Heater (CSTH) process (Ch. 4).
- Industrial Case-study: Gray-box model identification in a pulp mill (Ch. 5).

1.5.4 Additive Process FDI using continuous-time models

Representing continuous-time (CT) systems by discrete-time (DT) models is only an approximation of their dynamics. For example, many DT methods assume that signals in a system are piece-wise constant. This assumption can be true for manipulated variables that are driven by zero-order-hold (ZOH) devices. Other signals, e.g. disturbances and faults, are not always piece-wise constant. Therefore, DT fault models that attempt to associate the process behavior with the faults may not be correct. Additive process faults in linear dynamic systems are faults which directly affect the continuous-time states of the system in an additive fashion. Using CT gray-box identification techniques, it is sometimes possible to identify CT models which make the detection and isolation of additive faults in the states easier. We illustrate Continuous-time approaches to FDD and Fault Diagnosis Relevant Identification (FDRI) in this thesis using the following case-study:

- Subspace Identification of Residual Models for FDI in Continuous-time Systems: Simulation studies on a Quadruple-tank process (Ch. 6).

2

Steady-state model identification and fault diagnosis

2.1 Overview

Fault Detection and Diagnosis in complex processes is challenging, typically due to the lack of simplified empirical and first-principles models that can explicitly describe the process behavior. Modern plants contain large-scale, highly complex processes which operate with a large number of variables under closed loop control. In addition, the decrease in the price of instrumentation and data storage has led to a large increase in the amount of data logged in these plants. In this context, the use of multivariate data analysis tools has significant advantages in terms of identifying simple steady-state models which can then be used for process monitoring. Some of the multivariate statistical techniques which have made significant inroads into the process industry include Principal Components Analysis (PCA) and Partial Least Squares (PLS) Regression. These techniques have been used in a variety of applications including performance monitoring, building soft sensors, etc. In this chapter we give a brief description of these modeling techniques, the process monitoring methods based on these techniques and also study some of the issues which arise in the development of multivariate static models for industrial processes. These techniques are illustrated using simulation and industrial case-studies. We also present some of the challenges encountered in the development and online implementation of these techniques and the proposed solutions[1].

2.2 Steady-state model identification

In many chemical engineering applications, quality variables may not be measured as frequently as would be desired for satisfactory closed-loop control. For example, key product quality variables are available after several hours of lab analysis. Often, it is possible to estimate the quality variables using other process variables which are measured frequently. The relationship, or the model that is used to predict quality variables using other process variables is often called a *soft-sensor*. The quality-variable estimator is called a soft-sensor or a virtual sensor since it is based on software calculations rather than a physical instrument. The soft-sensors developed in this way can be used for inferential control or process monitoring. Discussions on inferential control can be found in (Kresta *et al.* 1994, Parrish and Brosilow 1985, Amirthalingam *et al.* 2000, Li *et al.* 2002).

Multivariate statistical techniques such as PCA and PLS have been applied for process monitoring, fault detection and static modelling in chemical processes (Kresta *et al.* 1991, Qin and McAvoy 1992, Qin 1993, Ricker 1988). Extensions of these approaches for handling dynamic and auto-correlated data have also been proposed (Ku *et al.* 1995, Lakshminarayanan *et al.* 1997). PLS and PCA are especially suited to handle data in which both the predictor variables (inputs) and the response variables (outputs) are measured with error. Under these circumstances, the ordinary least squares regression gives biased estimates for the parameters. Such a problem in which all the variables in the system are measured with error is termed an errors-in-variables (EIV) regression problem. These problems have been extensively studied in the statistical literature (Fuller 1987) and the problem statement reflects what is commonly encountered in practice.

PLS regression is a popular technique used in the development of soft-sensors in the form of static models for multivariate processes. The main advantage of using PLS for process modelling comes from its ability to decompose the problem of obtaining model coefficients from multivariate data into a set of univariate regression problems. Univariate regression is performed on latent variables obtained by projecting the input and output data onto directions along which the covariance between these variables is maximized. While PLS seems to work well in practice, the statistical properties of the PLS algorithms are not clear. The PLS estimate is a highly nonlinear function of the observed dependent variable. It has been proved in (Stoica and Söderström 1998) that PLS and PCA are equivalent to within a first-order approximation. However, an exact analysis is difficult because of the highly convoluted procedure employed in obtaining the PLS estimates.

---

[1] Sections of this chapter have been presented or published as:

1. Raghavan, H., S.L. Shah, R. Kadali and B. Doucette, "Monitoring Bitumen recovery in a separation cell using inferential sensors", submitted to the Canadian Journal of Chemical Engineering, July, 2003.
2. Raghavan, H., S.L. Shah, R. Kadali and B. Doucette, "Application of PLS-based Regression for monitoring Bitumen Recovery in a Separation Cell", In Proc. AdCHEM-2003, Hong Kong, Jan. 2004.

On the other hand, PCA is known to give consistent model estimates when the input and output variables of the system are measured with error. PCA is used for model estimation when the errors affecting the variables are independent and identically distributed (i.i.d.). This formulation is also known as Total Least Squares (TLS). Under these circumstances, the estimates can be considered to be maximum likelihood estimates, i.e., if all measurement error standard deviations have the same normal distribution. In practice, it is difficult to ensure that the errors are independent and identically distributed. Hence it might be advantageous to jointly estimate the model parameters and the error covariance matrix. This problem of joint estimation is well studied in the statistical literature (Gleser 1981, Chan and Mak 1983) and it is known that it is not possible to solve this problem without imposing additional restrictions on the structure of the error covariance matrix, due to lack of identifiability. One such constrained solution is provided by the Iterative PCA (IPCA) procedure (Narasimhan and Shah 2003).

An interesting yet disturbing result (Gleser and Hwang 1987) is that it is not possible to obtain confidence bounds on the estimated parameters for a fixed sample size, *no matter how large the sample size is*. This casts doubt on the usefulness of asymptotic results on the confidence intervals for the estimated parameters. To quote from Gleser and Hwang (1987): "*The* $100(1-\alpha)\%$ *confidence intervals for the estimated parameters are of infinite expected length and any confidence interval of fixed expected length has a confidence* $1-\alpha = 0$," i.e., the probability that the parameters will lie within that interval is zero, when the parameters are estimated from a sample of finite size. This is true irrespective of whether any distributional assumptions are made on the underlying signal. The result arises from the fact that by a suitable choice of the nuisance parameters (related to the variability of the underlying signal), the model can be made arbitrarily close to a model whose parameters are not identifiable. Hence it might not be possible to establish statistical properties other than consistency and some asymptotic properties such as normality and efficiency of the estimates for linear models in which the input and output variables are measured with error. This is true for the PCA, IPCA and the PLS procedures for static model identification, because they inherently assume that both the inputs and outputs are measured with error.

In spite of these problems, the IPCA procedure for soft-sensor design has some definite advantages. The model parameters obtained are scaling invariant and there is no ambiguity in the number of underlying components in the signal space when these are identified using the IPCA procedure. We now proceed to give a brief description of these techniques before illustrating these through case-studies.

2.2.1 Multiple Linear Regression (MLR)

Consider a data matrix $\mathbf{Z}$. Let us assume for the time-being that the variables in this data matrix can be separated into predictor variables $\mathbf{X} \in \Re^{N\times m}$ and response variables $\mathbf{Y} \in \Re^{N\times p}$ where N is the number of observations, m is the number of process variables and p the number of quality variables. Assume the presence of linear relationships between the predictor and response variables:

$$\mathbf{Y} = \mathbf{XC} + \mathbf{E} \tag{2.1}$$

Using the well known Ordinary Least Squares regression (OLS) we obtain the solution:

$$\hat{\mathbf{C}}_{ols} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \tag{2.2}$$

However, note that in the general Errors-in-variables (EIV) case, where both $\mathbf{X}$ and $\mathbf{Y}$ are measured with error, the Ordinary Least Squares procedure does not provide an unbiased estimate of the model parameters.

2.2.2 Errors-in-variables (EIV) Modeling

We represent the EIV case using the following model structure:

$$\begin{aligned} \mathbf{Y} &= \tilde{\mathbf{X}}\mathbf{C} + \mathbf{E}_1 \\ \mathbf{X} &= \tilde{\mathbf{X}} + \mathbf{E}_2 \end{aligned} \tag{2.3}$$

If $\mathbf{E}_2 = \mathbf{0}$, we can estimate $\mathbf{C}$ using Ordinary Least Squares (OLS) regression. In addition to this assumption, if there is a high degree of correlation within the $\tilde{\mathbf{X}}$ space, reduced rank regression techniques such as Principal Components Regression (PCR) and Canonical Variate Analysis (CVA) can be employed to solve this regression problem. Additionally there might be a high degree of correlation in the response space. Partial Least Squares (PLS) is one of the techniques used to deal with such data.

The Errors-in-variables (EIV) case assumes that $\mathbf{E}_2 \neq \mathbf{0}$. In this case, OLS and CVA give biased parameter estimates. This is illustrated in the following example:

Consider the system governed by the equation,

$$y = c\tilde{x} + e_1 \tag{2.4}$$

Let us assume that $\tilde{x}$ is measured with error, $x = \tilde{x} + e_2$. The OLS estimate for c is given by,

$$\hat{c} = (x^T x)^{-1} x^T y = ((\tilde{x} + e_2)^T (\tilde{x} + e_2))^{-1} (\tilde{x} + e_2)^T (c\tilde{x} + e_1) \tag{2.5}$$

Applying the expectation operator to this equation we see that $\hat{c}$ is not an unbiased estimate of c unless $e_2 = 0$ because, it depends on quantities such as $E(\frac{(\tilde{x} + e_2)^T (c\tilde{x} + e_1)}{((\tilde{x} + e_2)^T (\tilde{x} + e_2))})$, $E(e_2^T \tilde{x})$, $E(\tilde{x}^T e_1)$ and $E(e_2^T e_1)$.

Special techniques are necessary to handle the EIV regression case. These are especially important when dealing with data from the chemical industries because the measured data from various sensors naturally falls into this category. Hence techniques such as PLS and PCA have been proposed to handle the EIV and reduced rank regression problems.

We now describe briefly, the procedure used for steady-state model identification using these latent variable regression techniques.

2.2.3 Principal Components Analysis (PCA)

Consider the zero-mean, unit variance data matrix $\tilde{\mathbf{Z}} \in \Re^{N \times (m+p)}$, where N is the number of observations collected at time instants $t = 1$ to $t = N$, m is the number of process variables and p the number of quality variables.

A static relationship among these variables is given by,

$$\tilde{\mathbf{Z}}\Theta = \mathbf{0} \tag{2.6}$$

Note that the $\tilde{\mathbf{Z}}$ matrix contains both the process variables $\tilde{\mathbf{u}}(t)$ and the response or quality variables $\tilde{\mathbf{y}}(t)$. Hence,

$$\tilde{\mathbf{z}}(t) = \left[\begin{array}{cc} \tilde{\mathbf{y}}(t)^T & \tilde{\mathbf{u}}(t)^T \end{array}\right] \tag{2.7}$$

The matrix $\tilde{\mathbf{Z}}$ is obtained by stacking the $\tilde{\mathbf{z}}(t)$ vectors corresponding to the time instants $t = 1$ to $t = N$.

$$\tilde{\mathbf{Z}} = \begin{bmatrix} \tilde{\mathbf{z}}(1) \\ \tilde{\mathbf{z}}(2) \\ \vdots \\ \tilde{\mathbf{z}}(N) \end{bmatrix} \tag{2.8}$$

Performing an SVD on $\tilde{\mathbf{Z}}$ yields,

$$\tilde{\mathbf{Z}} = \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^T + \mathbf{U}_2\mathbf{S}_2\mathbf{V}_2^T \tag{2.9}$$

where, $\mathbf{U}_1$, $\mathbf{V}_1$ are the singular vectors corresponding to nonzero singular values and $\mathbf{S}_1$ is diagonal matrix with the singular values arranged along the diagonal in the order of decreasing magnitude. Similarly, $\mathbf{U}_2$, $\mathbf{V}_2$ are the singular vectors corresponding to zero singular values and $\mathbf{S}_2$ is diagonal matrix with the zero singular values. Since $\mathbf{V}_1^T\mathbf{V}_2 = \mathbf{0}$,

$$\tilde{\mathbf{Z}}\mathbf{V}_2 = \mathbf{0} \tag{2.10}$$

Hence $\mathbf{V}_2$ serves as an estimate for Θ. The observed data $\mathbf{Z}$ is related to $\tilde{\mathbf{Z}}$ by

$$\mathbf{Z} = \tilde{\mathbf{Z}} + \mathbf{E} \tag{2.11}$$

If $\mathbf{E} \sim N(\mathbf{0}, \mathbf{I})$, then PCA provides the Maximum Likelihood Estimate (MLE) of Θ. In addition, the number of static relationships can be estimated by inspecting the plot of the singular values. In this case, the dimensions corresponding to the unity singular values represent the residual space because the measurement noise has an identity covariance matrix. The dimensions corresponding to the singular values that are larger than unity represent the signal space. Once Θ has been estimated, it can be recast into a predictor form which can be used to predict the response variables as a function of the predictor variables.

2.2.4 Iterative Principal Components Analysis (IPCA)

Consider the case where $\mathbf{E}$ is not *i.i.d* but can be characterized by a Gaussian distribution with a covariance matrix $\mathbf{Q}$. In this case, one cannot determine the number of relationships in the data by looking at the singular values. Hence ad-hoc techniques are used for choosing the dimension of the signal space. Commonly used techniques include choosing the number of dimensions which explain 80% of the variation in the data or choosing dimensions for which the eigenvalues are greater than unity. Wentzell *et al.* (1997) proposed a maximum likelihood estimation technique for model identification using PCA when $\mathbf{Q}$ is known and the residual space dimension is specified. This technique is based on an alternating regression procedure which does not scale the data. Instead, it iteratively transforms the model identified by PCA on un-scaled data, until a maximum likelihood estimate is obtained. A simpler method to obtain the maximum likelihood estimate of Θ is by scaling the data using the matrix $\mathbf{L}^{-1}$, where $\mathbf{Q} = \mathbf{L}\mathbf{L}^T$. For example, $\mathbf{L}$ could be the cholesky factor of $\mathbf{Q}$. One of the advantages of scaling the data using $\mathbf{L}^{-1}$ is that the number of static relationships can be estimated by inspecting the plot of the singular values as in PCA.

In practice, it is unrealistic to expect a priori knowledge of the covariance matrix $\mathbf{Q}$. However, it may be possible to jointly estimate the covariance matrix and the static relationships from the data if the data satisfies certain identifiability restrictions (Chan and Mak 1983). For example, if the covariance matrix is diagonal, the identifiability restriction can be expressed as, $\frac{r(r+1)}{2} \geq (m+p)$, where r refers to the dimension of the residual space. The Iterative PCA (IPCA) procedure for joint estimation of the static model and the error covariance matrix is based on constructing the likelihood function of the constraint residuals, and determining the estimates of the model and error covariance matrix which maximize this likelihood function. The algorithm starts with an initial estimate of the model parameters ($\hat{\Theta}_0$) which is obtained using PCA. Using this estimate for the model, the constraint residuals are obtained from the data following:

$$\mathbf{r}(t) = \mathbf{z}(t)\hat{\Theta}_0 \tag{2.12}$$

Here we assume that $\mathbf{z}(t) \in \Re^{1\times(m+p)}$ is the t^{th} observation. Using these constraint residuals, the error covariance matrix is estimated as the solution of the following nonlinear optimization problem, where $|\,.\,|$ refers to the determinant.

$$\hat{\mathbf{Q}} = \arg\min_{\mathbf{Q}} N \log|\hat{\Theta}_0^T \mathbf{Q} \hat{\Theta}_0| + \sum_{t=1}^{N} \left(\mathbf{r}(t)^T \left(\hat{\Theta}_0^T \mathbf{Q} \hat{\Theta}_0 \right)^{-1} \mathbf{r}(t) \right) \tag{2.13}$$

Following this estimation, the data is scaled using the inverse of the cholesky factor of the estimated covariance matrix. PCA is applied on the scaled data and the constrained residuals recalculated. This procedure is repeated until convergence. Once $\hat{\Theta}$ has been estimated using the IPCA procedure, it can be recast into a predictor form and provide maximum likelihood estimates of the states and the response variables.

2.2.5 Partial Least Squares (PLS)

Because of the high degree of correlation among the variables within the predictor space the matrix $\mathbf{X}^T\mathbf{X}$ may be ill-conditioned leading to high variance of the model parameters identified using the OLS procedure. In addition we may be interested in obtaining the directions, along which the common (second-moment) information between these blocks is concentrated. To satisfy these objectives, the following procedure is adopted in PLS regression. The matrix $\mathbf{X}$ is decomposed into a scores matrix $\tilde{\mathbf{T}} \in \Re^{N\times a}$ and a loadings matrix $\tilde{\mathbf{P}} \in \Re^{m \times a}$, where a is the number of PLS components used. Hence the following decomposition is achieved:

$$\mathbf{X} = \tilde{\mathbf{T}}\tilde{\mathbf{P}}^T + \tilde{\mathbf{F}} \tag{2.14}$$

where $\tilde{\mathbf{F}}$ is a residual matrix. Similarly $\mathbf{Y}$ is decomposed as

$$\mathbf{Y} = \tilde{\mathbf{U}}\tilde{\mathbf{Q}}^T + \tilde{\mathbf{G}} \tag{2.15}$$

To obtain the loadings vectors the following algorithm is used:

1. Initialize, $\mathbf{Y}_1 = \mathbf{Y}$ and $\mathbf{X}_1 = \mathbf{X}$ and $i = 1$.

2. Perform a Singular Value Decomposition (SVD) on $\mathbf{X}_i^T\mathbf{Y}_i$ and calculate $\mathbf{j}_i$, the left singular vector corresponding to the largest singular value ω_i and $\mathbf{q}_i$, the corresponding right singular vector. This SVD calculation corresponds to capturing the direction ($\mathbf{j}_i$,$\mathbf{q}_i$) that maximizes covariance between $\mathbf{X}_i$ and $\mathbf{Y}_i$.

3. Let $\mathbf{t}_i$ and $\mathbf{u}_i$ be the corresponding scores. Perform a univariate regression between $\mathbf{t}_i$ and $\mathbf{u}_i$ to obtain $\mathbf{b}_i$.

4. The loadings vector for $\mathbf{X}_i$ is given by

$$\mathbf{p}_i = \frac{\mathbf{X}_i^T\mathbf{t}_i}{\mathbf{t}_i^T\mathbf{t}_i} \tag{2.16}$$

5. Deflate $\mathbf{Y}$ and $\mathbf{X}$ according to

$$\begin{aligned} \mathbf{Y}_{i+1} &= \{\mathbf{Y}_i - \mathbf{b}_i\mathbf{t}_i\mathbf{q}_i^T\} \\ \mathbf{X}_{i+1} &= \{\mathbf{X}_i - \mathbf{t}_i\mathbf{p}_i^T\} \end{aligned} \tag{2.17}$$

6. Set $i = i + 1$.

7. Go to step 2.

After a stages, the approximations are

$$\begin{aligned} \mathbf{X} &\approx \mathbf{t}_1\mathbf{p}_1^T + \mathbf{t}_2\mathbf{p}_2^T + \cdots + \mathbf{t}_a\mathbf{p}_a^T \\ \mathbf{Y} &\approx \mathbf{u}_1\mathbf{q}_1^T + \mathbf{u}_2\mathbf{q}_2^T + \cdots + \mathbf{u}_a\mathbf{q}_a^T \end{aligned} \tag{2.18}$$

Hence we get the PLS estimate of the model coefficients as:

$$\hat{\mathbf{C}}_{pls} = \tilde{\mathbf{J}}(\tilde{\mathbf{P}}^T\tilde{\mathbf{J}})^{-1}\mathbf{B}\tilde{\mathbf{Q}}^T \tag{2.19}$$

where, the columns of $\mathbf{J}$ and $\tilde{\mathbf{Q}}$ contain the singular vectors of the SVD's carried out at each stage, the columns of $\tilde{\mathbf{P}}$ contain the loadings vectors of the $\mathbf{X}$ matrix and $\mathbf{B}$ is a diagonal matrix containing the latent variable regression coefficients from each stage.

2.3 Bitumen Recovery Prediction

In this section, an industrial example of the application of the PLS regression and IPCA approaches for the development of inferential sensors to predict the Bitumen Recovery in a separation cell is shown. Some of the challenges encountered in the development and online implementation of the inferential sensors and the proposed solutions are presented.

In many processes product quality is estimated through lab assays. These quality variables are infrequently and often irregularly sampled. In contrast, other process variables are sampled frequently. Inferential or "soft" sensors can be used for predicting the inter-sample behavior of the quality variables. A soft sensor is a mathematical model that describes the relationship between quality variables and process variables. We present an industrial example of the development of inferential sensors to predict the Bitumen Recovery in a separation cell.

2.3.1 Process description

We have developed soft-sensors to predict the Bitumen Recovery in a separation cell at Suncor Energy's Extraction facility at Fort McMurray in Alberta, Canada. The separation cell is used in the extraction of bitumen from oil sands. Oil sands are deposits of bitumen, that must be treated to convert them into crude oil which can then be refined in conventional refineries. The main processes in converting the oil sands to crude oil are Mining, Extraction and Upgrading. In the mining stage, the oil sands are mined using trucks and shovels. This is followed by the extraction stage in which bitumen is separated from the sand using processes such as froth-flotation. The bitumen is then converted to crude oil in the upgrading stage.

The extraction operations can be briefly described as follows: The oil sand is first passed through a slurry preparation stage. The main operation in this stage is to form a slurry using hot water, oil sands and caustic. Heat is used to reduce the viscosity of the bitumen. Caustic helps in the attachment of bitumen to the air in the froth formation while releasing it from the sand particles. The bitumen then forms small globules that are important in the formation of froth. Agitation also aids in breaking up the oil sand. The slurry passes through a series of vibrating screens that separate and reject any rocks or clumps of clay still present in the slurry. It is then pumped into separation cells.

A schematic of a separation cell is shown in Fig. 2.1. The separation cell allows the slurry to settle out into its various layers, the most important layer being the froth layer which rises to the top. The tailings sand sinks to the bottom. The middle layer is called the middlings layer and consists of bitumen, clay and water. The middlings remain suspended between the sand and the bitumen froth until it is drawn off and sent through the secondary separation cell. The secondary separation vessel extracts the remaining bitumen from the middlings.

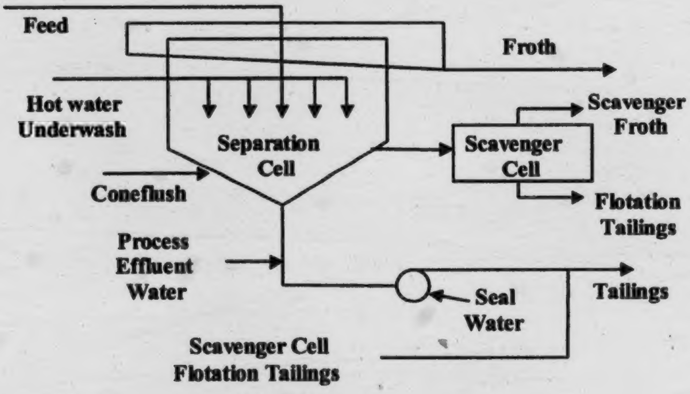

Feed
Froth
Hot water Underwash
Separation Cell
Scavenger Cell
Scavenger Froth
Coneflush
Flotation Tailings
Process Effluent Water
Seal Water
Tailings
Scavenger Cell Flotation Tailings

Figure 2.1: Process Flow-sheet for Separation Cell

The main objective in the operation of the separation cell is to maximize the amount of bitumen in the froth and minimize the amount of bitumen lost in the tailings and middlings streams. A measure of the efficiency of operation of the separation cell is given by the Bitumen Recovery which can be calculated from the predictions of quality variables using the following equation:

$$Rec = \frac{F_{fr}\rho_{fr}C_{fr}}{F_{fr}\rho_{fr}C_{fr} + F_t\rho_tC_t + F_{ft}\rho_{ft}C_{ft}} \tag{2.20}$$

where, *Rec* is the Bitumen Recovery in the cell, F_{fr}, F_t and F_{ft} refer to the Froth, Tailings and Flotation Tailings flows, ρ_{fr}, ρ_t & ρ_{ft} refer to the densities of Froth, Tailings and Flotation Tailings and C_{fr}, C_t & C_{ft} refer to the concentrations of Bitumen in the Froth, Tailings and Flotation Tailings in wt% respectively. Hence the quality variables of interest are concentrations of Bitumen in the Froth, the Tailings and the Flotation Tailings. In our soft-sensor development, we have used 25 process variables, measured every minute, to predict these 3 quality variables. Of the three product variables, the froth bitumen concentration is available through lab analysis every 12 hours while the tailings and flotation tailings bitumen concentrations are available every 2 hours.

2.3.2 Challenges in soft-sensor development

Monitoring the extraction of bitumen from oil sands is a problem which poses some unique challenges. These include, in the words of a practicing engineer from this industry, "*changing process conditions, wide operating regions, bad data and lack of good software resources*". In addition, we have encountered other challenging problems for which we have some suggested solutions. Many of these solutions may also apply to other applications.

2.3.3 Data Acquisition and Import

Many publications on online applications of process monitoring and soft-sensor development for industrial processes start with statements such as: *Assume that the data is available in a matrix* X. However, it should be realized that there is a large amount of data preprocessing required before the data from an industrial process can be made available in a form that can be used by routines which are used to build monitoring and soft-sensor applications. In fact, it would be fair to say that more than 80% of the time required to develop online monitoring and soft-sensor applications is spent in data preprocessing and data quality analysis.

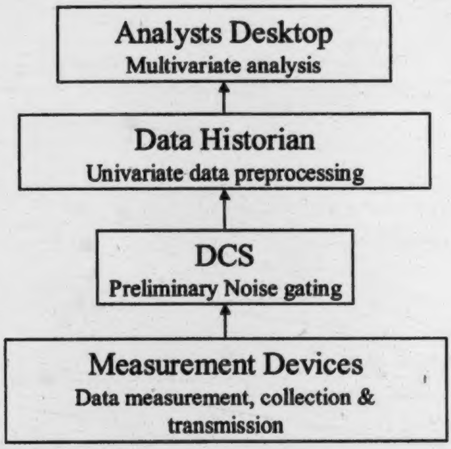

Analysts Desktop
Multivariate analysis
Data Historian
Univariate data preprocessing
DCS
Preliminary Noise gating
Measurement Devices
Data measurement, collection & transmission

Figure 2.2: Data-based view of an industrial process

A Data-based view of a typical industrial process is shown in Fig. 2.2. Data is collected by the field instruments, typically asynchronously, and is then transmitted to the Distributed Control System (DCS). Some preliminary data preprocessing is performed before or at the

DCS level. This includes, noise gating, setting high and low limits and quantization which may be introduced due to data storage constraints. The DCS is used for online display of the current state of the process (univariate display of some key variables) and hence online control and monitoring operations which could be automatic or manual (operator driven). The length of data available in the DCS is of the order of a few days. It is of high quality in terms of the information it possesses about the process. This is because, the sampling rates in the data are constrained only by the sampling rates of the physical instruments themselves. However, this data is not suitable for developing monitoring or soft-sensor applications because of its short length which results in its inability to capture all the deterministic and stochastic modes of the process. For long term applications, data is stored in historical databases using software appropriately termed *Plant Historians*. Historians have a number of options which can be used to minimize the number of data points stored in order to make the best use of the data storage facilities, *i.e.*, data is compressed for storage efficiency. However, most of the historian software available currently, treat data in a univariate fashion. Hence it is important to check the fidelity of the archived data from a multivariate analytic perspective before proceeding with such analysis. After such a preliminary analysis, if it is found that there is indeed a significant loss of information in the archival process, the parameters affecting the data storage in the historian, *i.e.*, the compression factors, should be changed. It will take some time before enough data is collected which can be then used for multivariate analysis.

2.3.4 Data Quality Assessment

There are a number of ways in which plant historians process data before storing it. While these processes are useful in improving the quality of data from a univariate display point-of-view, some of these processes are harmful for multivariate data analysis because they introduce non-linear relationships, which may alter our inferences when we use the processed data. Hence it is important to note these settings and change some of them before we proceed to perform tasks such as building static or dynamic soft-sensors or monitoring schemes using such data. Some of the pre-processing steps include, outlier detection and replacement, determination of extent of data compression, estimating quantization effects and limits, low pass filtering and smoothing.

Outlier detection and removal: The objective of outlier detection and removal is to remove random data spikes that result from instrument or communication noise. In general, univariate routines are applied on data to remove statistical outliers. Some common routines currently used in historians are based on linear regression and standard deviation calculations based on small samples of data. For example, a common technique involves selection of a fixed small number of samples of data before and after a particular data point, performing linear regression using these surrounding values, calculating the standard deviation of the residuals and judging whether the data point in question is within a specified number of standard deviations from the *line of best fit*. If it is outside this limit, it is considered as an outlier and replaced by the regression estimate. It is clear from this description that the method is highly nonlinear and harmful from a multivariate point of view because it does not take true multivariate nature of the process into account. In addition if the historian is set in such a way that it processes variables at a common base sample rate (this is generally the case and this rate is around 1 min in many cases), then the dynamic time-scales of the individual variables is ignored in this procedure. Hence it might be better to turn off univariate outlier detection and replacement functions in the historian when data collection for multivariate analysis is being performed.

Data compression: Historical data contains a lot of useful information. However, it is difficult to store and retrieve large quantities of data because of hardware and network bandwidth requirements. Hence data compression is used to minimize the amount of data stored in a historical archive.

In practice historians use a variety of data compression techniques, e.g., box car, backward slope, swinging door etc. to compress the data. A number of the compression algorithms currently used in historians are based on piecewise linear interpolation. For example, the swinging door algorithm (Inc 2002) which is a commonly used algorithm is illustrated in Fig. 2.3. A detailed description of the methodology is available in (Thornhill *et al.* 2004). The points 'a' and 'b' are obtained by adding and subtracting the compression deviation from the recorded point 'c'. The top line is drawn through 'a' and the data point which maximizes the magnitude of the slope. The bottom line is drawn through 'b' and the data point which minimizes the magnitude of the slope. The middle dashed line is obtained by connecting the last recorded point to the new value. If the middle line has a smaller slope than the top line or a greater slope than the bottom line, the previous value is recorded. Fig. 2.3 shows that a data point being recorded because the top slope is greater than the middle slope. The reconstructed data is obtained by linear interpolation between recorded points.

While these might be useful from a data storage point of view, they adversely affect the results of multivariate analysis. A recent study (Thornhill *et al.* 2004) concludes that compression induces changes to many of the quantities commonly used in data-driven process analysis. It has a significant impact on the basic statistical properties (like mean and

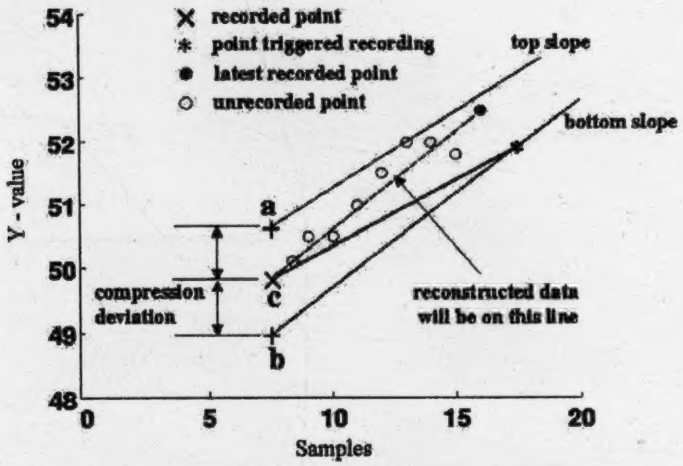


Figure 2.3: Swinging Door algorithm for data compression

variances) and inferences obtained from control loop performance assessment and process monitoring studies. Hence it is necessary to turn off univariate data compression algorithms in the historian when data collection for multivariate analysis is being performed.

Quantization & Data limits: Historians have upper and lower limits on the values of the stored variables. These are sometimes called *high extreme* and *low extreme* values. These are artificially introduced in the tag definition and are potentially harmful for data analysis, if the physical variable reaches these limits. However, these can be set to meaningful values, like the range of the instrument. In some cases, due to economic considerations, the production rate may be more than the rate the instrument was designed for. Hence the sensor readings might be getting truncated at the high and low ends of the range of the instrument.

Fig. 2.4 shows one such example for a flow rate. This problem has been studied in statistical literature and is termed *data censoring* (Gupta 1952). For a sensor which is subject to a high degree of censoring, it might be necessary to disregard the sensor reading in the analysis. In case the data from the sensor is important for the model development, it might be necessary to use inferential strategies to predict the values of the sensor reading when it gets censored. The expectation maximization (EM) algorithm (Dempster *et al.* 1977) provides one such inferential strategy.

Data quantization is the minimum step size used for discretization in the data. This is generally a function of the data storage allotment in the historian i.e., the number of

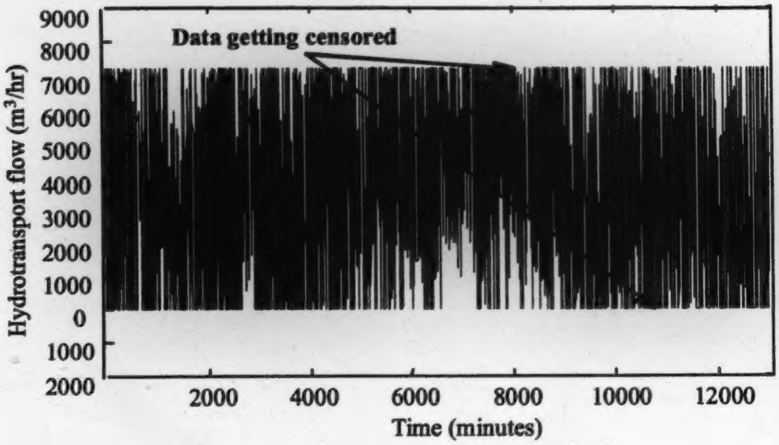


Figure 2.4: Example of a flow tag getting censored

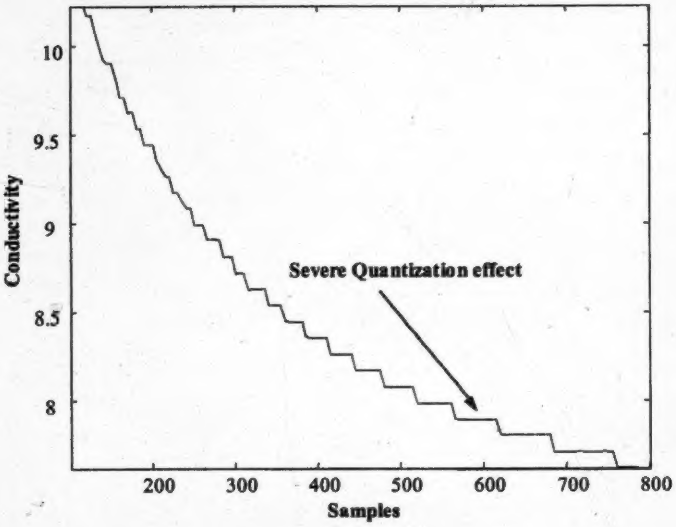


Figure 2.5: Example of severe quantization in a conductivity tag

bytes allotted to each data point and the physical capabilities of the instrument. Given the current trend of relatively inexpensive methods of data storage, data quantization is generally not a problem. However, it is good to find this out if the data seems to be overly quantized because this might affect the results of the analysis. Fig. 2.5 shows an example of a conductivity tag suffering from quantization problems due to physical limitations in the measurement. If one does find out that a particular variable behaves in a discrete fashion because of physical or storage limitations, it might be necessary to resort to discrete or hybrid regression techniques. If the problem is with data storage, for example floating point numbers being saved as scaled 16 bit integer values rather than 4 byte floating representation because of the reduction in data storage requirement, it might be necessary to turn this feature off when data collection for multivariate analysis is being performed.

Data smoothing and Low-pass filtering: In some historians, data smoothing or low-pass filtering is performed to remove the effect of random noise which is typically concentrated in high frequency regimes. While this operation improves the appearance of data from a univariate perspective, it might modify the results obtained from multivariate analysis. For example, let us consider a system in which there is a static relationship between an input and output variable $y(t) = Ku(t)$. If a filter $F(q^{-1})$ is applied to the input $u(t)$, the relationship between the filtered input $u_F(t)$ and the output is different from the original relationship. It might be still possible to obtain the true relationship between input and the output, if the same filter is applied on the output as well. We will obtain a consistent estimate for K using the filtered data, though the statistical properties of the estimate (particularly the second order properties) may not be the same. Hence it is necessary to carefully choose the smoothing parameters for the historian while identifying relationships from multivariate data.

Sample consolidation: One of challenges encountered while developing soft-sensors is due to the practice of physical consolidation of samples of the quality variables. The reason for this practice is to obtain representative samples of the quality variables. The procedure involves mixing a number of physical samples of the product collected at different time instants before performing lab analysis. For the process under consideration, consolidation is achieved using a flow totalizer and a triggering mechanism. When the cumulative flow in a line exceeds a set point it sets off a mechanism which leads to the collection of a sample in a container. The consolidation mechanism is illustrated in Fig. 2.6. This process continues for about 12 hours for the froth sample and about 2 hours for the other quality variables. At the end of the collection, the container has a mixture of the samples collected over this period. This liquid is then stirred for homogeneity and the consolidated sample is used for analysis. In order to build realistic models using such samples, it is important that the modelling methodology including the data pre-treatment mimic the process as much as possible. Hence we resorted to time-averaging of the input data as dictated by the sample consolidation mechanism before the actual regression was performed.

Let us assume that k samples are collected at times $T_1+t_1, T_1+t_2, ..., T_2 = T_1+t_k$, where

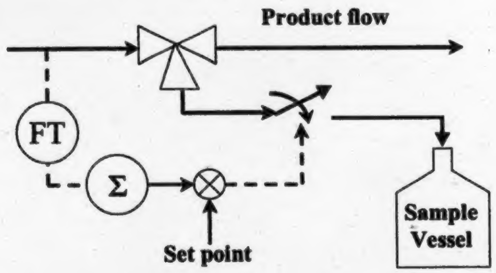


Figure 2.6: Sample Consolidation mechanism

T_1 and T_2 refer to times when the vessel is removed for analysis and $t_1, t_2, \ldots, t_k$, refer to the times when the trigger mechanism is engaged. Then, assuming that equal volumes of the product are sampled at the sample instant, the following equation holds approximately:

$$Y_{av} \approx \frac{1}{k} \sum_{t_i=t_1}^{t_k} Y(t_i)$$

Under the assumption that the process can be represented well using a linear static-plus-time delay model of the form,

$$Y(t_i) = a_1 u_1(t_i - t_{d1}) + a_2 u_2(t_i - t_{d2}) + \ldots + a_m u_m(t_i - t_{dm})$$

where, $a_1, \ldots, a_m$ are the static regression coefficients of the m input variables $u_1, \ldots, u_m$ and the d_i is the time delay between the i^{th} input and the output, we get the expression:

$$Y_{av} \approx \frac{1}{k}\{a_1 \sum_{t_i=t_1}^{t_k} u_1(t_i - t_{d1}) + \ldots + a_m \sum_{t_i=t_1}^{t_k} u_m(t_i - t_{d1})\}$$

Hence time-averaging can be used to mimic the sample consolidation mechanism.

Effect of large sampling intervals on data size: Another challenge is in the large sampling times for the quality variable. The economic cost of performing lab tests to obtain concentrations, is one of the reasons for the large sampling intervals. In addition, once soft-sensors are in place to predict important quality variables, the plant personnel tend to use these to increase the sampling intervals in order to save money on lab tests. The sampling time for the froth bitumen is 12 hours. This means that even data collected over the course of a few months would yield very few values for the froth bitumen. For example we obtained only 60 samples over 30 days. In addition, the ratio of sampling time of the quality variable to that of the process variables is 720. Developing multi-rate models with such large sampling ratios given that we have 25 inputs, is not practical.

For static regression problems, where we are interested in capturing spatial relationships between different variables rather than temporal relationships, we can use the data at the slow sample rates. We have adopted this procedure for the models developed in this exercise. This reduces the number of samples available for modeling. This problem is especially important when considering more complicated nonlinear structures like neural-networks in the soft-sensors. Apart from the black-box nature of neural network models which impacts the interpretability of the models, the relatively few number of data points makes the coefficients used in the model unreliable. Hence we resorted to static-plus-delay structures where the optimal delay was obtained using physical quantities like residence time.

Estimating time delays in industrial processes: The problem of time-delay estimation was found to be particularly challenging. In many process monitoring applications, we are interested in developing static-plus-delay models in spite of the presence of dynamic elements because of the ease of identification and implementation. In addition, in a number of soft-sensor applications, it is not easy to identify dynamic models because of the unavailability of lagged output values in contrast to the availability of lagged input values. While identifying these models from routine operating data, it may be better to lag the process variables using transport delays, known *a priori* from process knowledge, rather than estimating time delays from the data, if we are interested in identifying plant relationships rather than correlations which arise from operational and control strategies. Transport delays can be estimated based on physical locations of the process elements and are functions of the production rate. However, we can make the reasonable assumption that these transport delays can be approximated as constants. In addition, care should be taken to include recycle effects when transport delays are chosen. In the case of material or energy recycle, the model structure should be expanded to accommodate these effects. One question that arises in the model identification is the optimal method for handling the dynamic elements given the fact that we are interested in identifying static-plus-delay models. This question is related to under-modeling and can be solved using a Prediction Error Minimization Method (PEM) framework. We present the optimal delay for a Single-Input-Single-Output (SISO) process. This result can be extended to a Multiple-Input process in a straightforward manner.

PEM approach for selection of time delay: Consider a linear, time invariant (LTI)

SISO process that can be represented using its impulse response function (Ljung 1999).

$$\begin{aligned} y(t) &= G(q^{-1})u(t) + v(t) = \sum_{k=1}^{\infty} g(k)u(t-k) + v(t) \\ y(t) &= g(1)u(t-1) + g(2)u(t-2) + \ldots + g(k)u(t-k) + \ldots + v(t) \end{aligned} \tag{2.21}$$

where, u refers to the input, y refers to the output and v refers to unmeasured disturbance. In the following discussion, we make the assumption that $v(t)$ is a white noise process. The optimal delay estimate when the system is subjected to colored disturbances is beyond the scope of this thesis.

We are interested in estimating a static-plus-delay predictor for the process shown in Eq. 2.21.

$$\hat{y}(t) = Ku(t-d) \tag{2.22}$$

The objective is to minimize the prediction error.

$$\hat{\theta} = \arg\min_{\theta} \left\{ J = \sum_{t=1}^{N} (y(t) - \hat{y}(t))^2 \right\} \tag{2.23}$$

where, $\theta = \{K, d\}$. Let us assume for a moment that the input to the process is white noise so that,

$$R_u(\tau) = \overline{E}u(t)u(t-\tau) = \alpha\delta_{\tau 0} \tag{2.24}$$

where the $\overline{E}$ operator is defined as follows (Ljung 1999):

$$\overline{E}\{\ \} = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \{\ \} \tag{2.25}$$

and δ is the kronecker delta function. It is easy to see in this case that the optimal value for d is obtained when it corresponds to the maximum impulse response coefficient.

$$\begin{aligned} J &= \overline{E}\left\{(y(t) - \hat{y}(t))^2\right\} \\ &= \overline{E}\left\{(g(1)u(t-1) + g(2)u(t-2) + (g(d) - K)u(t-d) + \ldots)^2\right\} \\ &= g(1)^2 + g(2)^2 + \ldots + (g(d) - K)^2 + \ldots \end{aligned} \tag{2.26}$$

In order to minimize J, we have to choose K to be equal to the largest impulse response coefficient. Hence it is clear that the optimal choice of the delay depends on the impulse function of the system.

If the input is not white, a pre-whitening filter $W(q^{-1})$ can be designed so that $u(t) = W(q^{-1})e(t)$, where $e(t)$ is a white noise sequence (Ljung and Glad 1994). In this case, the optimal choice of the delay is a function of the impulse response coefficients of the filter $F(q^{-1}) = G(q^{-1})W(q^{-1})$.

We now present an example of optimal static-plus-delay approximation for a first order, delay-free process whose time constant is 15 seconds. The system is subjected to a random binary input. The frequency band for the input's frequency contents is chosen as $[0,\omega]$, expressed in fractions of the Nyquist frequency (See the *idinput* command in MATLAB©). We modify the frequency characteristics of the input by reducing the value of ω, to see its effect on the optimal time delay estimate.

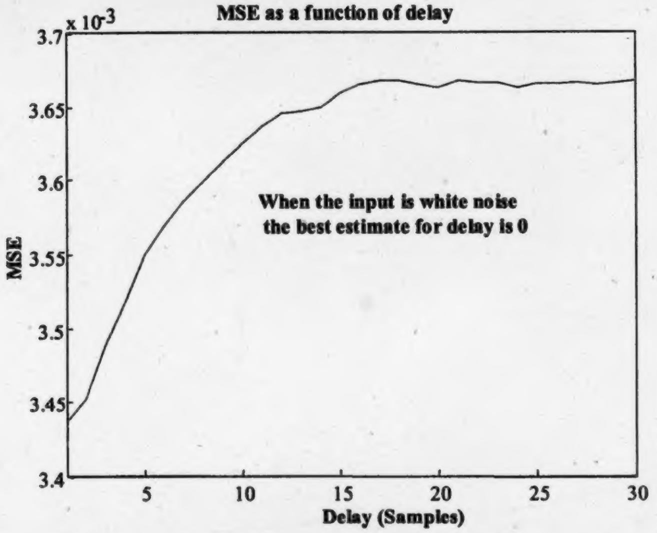


Figure 2.7: MSE as a function of delay when $\omega = 1$

Fig. 2.7 shows the Mean Squared Error (MSE) as a function of the delay when $\omega = 1$. The optimal delay has been identified as 0. Fig. 2.8 shows the MSE values for $\omega = 1, 0.8, 0.6, 0.4$ and 0.2 and Fig. 2.9 shows the MSE values for $\omega = 0.2$. It is clear that the optimal delay value tends to move towards the time constant.

Hence it may be concluded that for a system that is subject to inputs which have most of their power in the low frequency regime, the optimal choice of the delay is a function of the time constant. In practice, approximate residence times may be known for the dynamic elements and these may be used to lag the process variables if these elements are known to possess low order dynamics. We have used transport delays and residence times to lag the variables in the Bitumen recovery soft sensors developed in this project.

Nonlinear transformations:

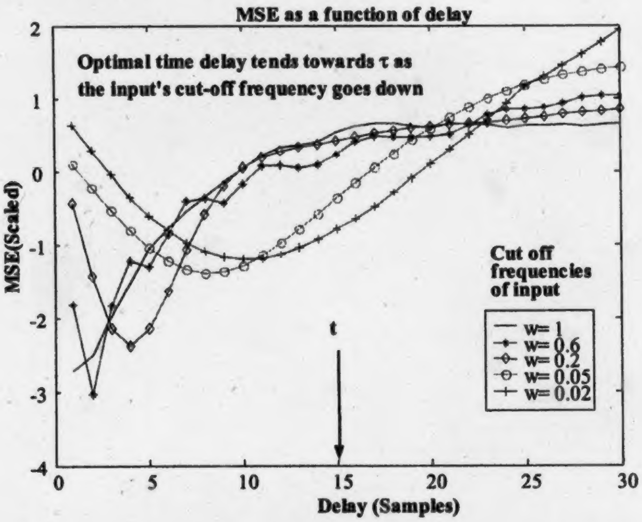


Figure 2.8: MSE as a function of delay for various ω values

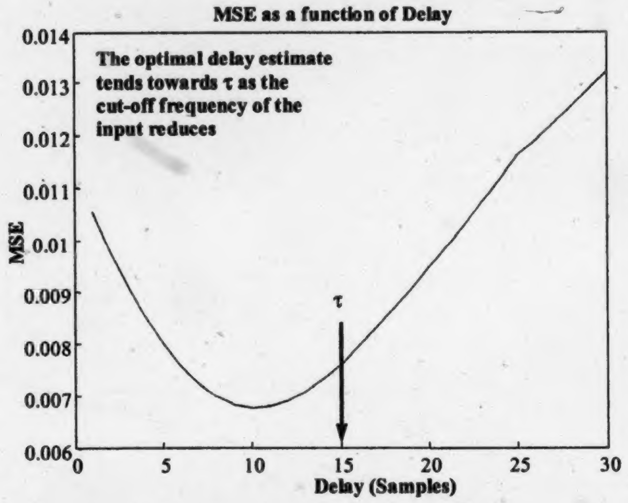


Figure 2.9: MSE as a function of delay when $\omega = 0.2$

While developing models of systems using linear regression, it is desirable to have normally distributed errors affecting the system and a linear relationship between the variables in the system. However, in practice these conditions may not hold. For example, the presence of a nonlinear relationship between the dependent and independent variables, or non-normality of the independent variables or the error manifests itself as non-normality of the dependent variable. Hence it is important to check whether it might be inappropriate to identify a standard linear model using a given set of data. If nonlinearity is suspected, we may need to use suitable transformations of the variables to coax the dependent variable to normality or to produce a linear relationship between X and Y. A dependent variable may not be normally distributed if its values are bounded, creating a skewed distribution. When it comes to inference of parameters from regression, it is important to ensure that the errors are normally distributed. A non-normal dependent variable does not necessarily mean a non-normal distribution of errors. However, the converse is often encountered. This argument is also supported by the common practice of drawing conclusions about the error distribution from the distribution of the residuals. When the dependent variable is found to be non-normal, one may consider using transformations to normalize the dependent variable. A few common transformations that can be used for dependent variables, include the logarithmic $(Z = log(Y))$, exponential $\left(Z = e^Y\right)$, power $(Z = Y^p)$ and logistic $\left(Z = \frac{log(Y)}{1-log(Y)}\right)$ transformations.

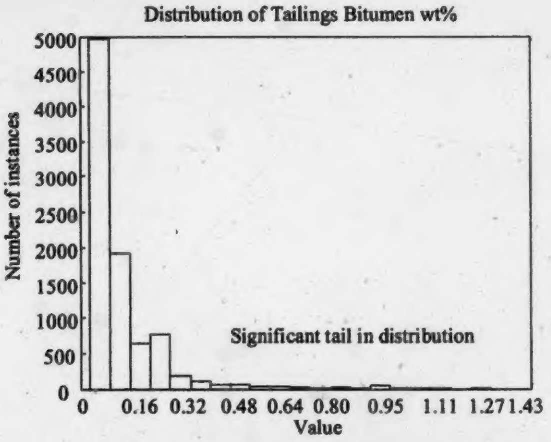


Figure 2.10: Tailings distribution shows significant deviation from normality

For the Bitumen recovery separation cell, the distributions of two of the quality variables show significant deviation from normality. They are the Bitumen concentrations in the Tailings and Flotation tailings. The distribution of the Tailings Bitumen is shown in Fig. 2.10 to illustrate this. These quality variables take non-negative values which are generally low, except during upsets, which are characterized by large spikes in these variables. Performing linear regression without transformation leads to poor prediction

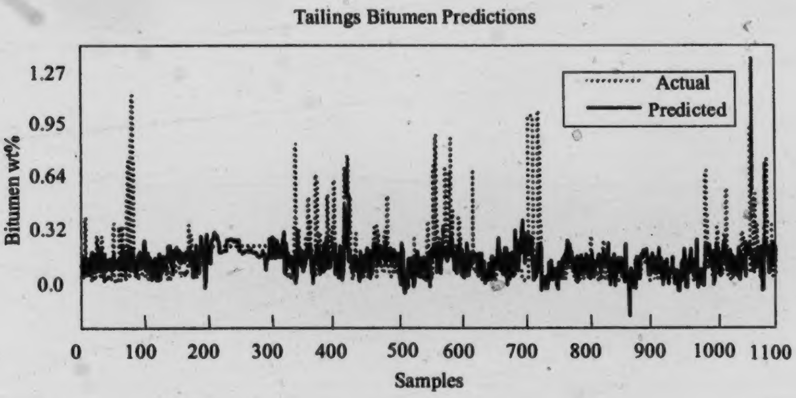


Figure 2.11: Linear model does not predict spikes adequately

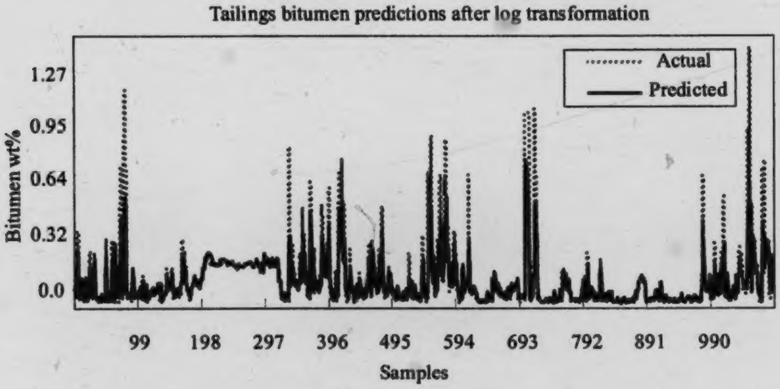


Figure 2.12: Predictions improve after logarithmic transformation

of these spikes (Fig. 2.11). It is important for the soft-sensor to predict spikes when they occur because these represent loss of bitumen into the tailings streams. Predicting the exact value of the spikes is relatively unimportant. Due to the nature of the distribution the logarithmic transformation has been applied on these dependent variables which led to a significant improvement in the quality of the predictions (Fig. 2.12) .

2.3.5 Online Results

Soft-sensors were developed for predicting the bitumen concentration (in weight%) in the froth, middlings and tailings streams using data collected over a period of three months. The soft-sensors used 25 measurements based on the process variables shown in table 2.1.

Table 2.1: Process variables used in soft-sensor

Feed flow
Feed density
Froth density
Middlings density
Flotation Tails density
Underwash
Cone flush
Tailings dilution flow
Dilution water flow
Middlings flow
Scavenger tails flow
Froth flow
Tailings flow
Tailings pump amps
Froth level
Tailings pump discharge pressure
Dearator pump amps

The data was split into two portions, training data which was used for developing the model and validation data, which was used to test the model. The results of PLS and IPCA predictions were compared using the Mean Squared Error (MSE) and Correlation Coefficient (CC) performance indices. These are defined as follows:

$$MSE = \sqrt{\frac{1}{N}\left(Y - \hat{Y}\right)^T\left(Y - \hat{Y}\right)} \tag{2.27}$$

where, Y refers to the lab measurement of the quality variable and $\hat{Y}$ refers to the prediction using IPCA or PLS.

$$CC = \frac{Cov(Y, \hat{Y})}{\sqrt{Var(Y)Var(\hat{Y})}} \tag{2.28}$$

where, Y refers to the lab measurement of the quality variable and $\hat{Y}$ refers to the prediction using IPCA or PLS, Cov and Var stand for covariance and variance respectively.

The advantage of using IPCA for developing the soft-sensor is that the number of dimensions of the signal space can be easily determined by looking at the plot of the singular values. The singular values corresponding to the signal space are greater than unity and those corresponding to the residual space are equal to unity. This is illustrated in Fig. 2.13 where a portion of the singular value plot is shown. The corresponding

Table 2.2: *Singular values obtained using the IPCA procedure*

Factor	Singular Values	Expected Range
1	39442	> 1
2	2625.6	> 1
3	1043.8	> 1
4	176.5	> 1
5	66.404	> 1
6	36.484	> 1
7	8.8514	> 1
8	6.3739	> 1
9	3.2607	> 1
10	2.8921	> 1
11	2.0172	> 1
12	1.5626	> 1
13	1.3656	> 1
14	1.0648	$= 1$
15	1.0146	$= 1$
16	1.0058	$= 1$
17	0.99491	$= 1$
18	0.97516	$= 1$
19	0.94042	$= 1$

Table 2.3: *Performance indices for the soft-sensors*

Performance Indices		
Tailings Soft Sensor		
	IPCA	PLS
MSE	0.1007	0.1171
CC	0.7202	0.5913
Flotation Tails Soft Sensor		
	IPCA	PLS
MSE	0.2983	0.3757
CC	0.7221	0.5983

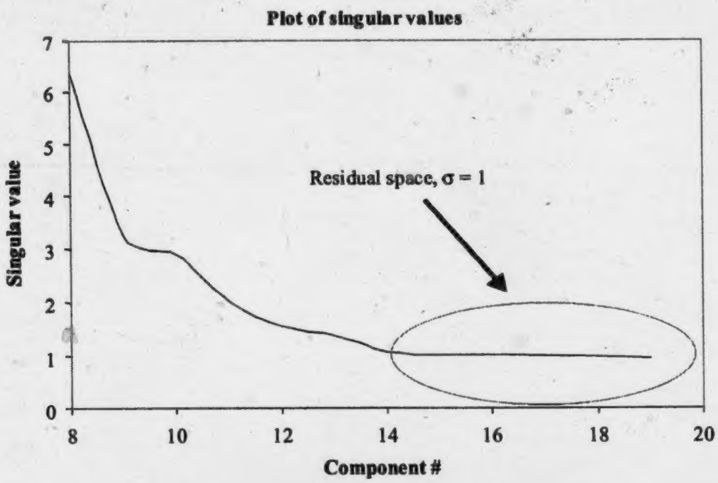


Figure 2.13: Plot of the singular values obtained through IPCA procedure

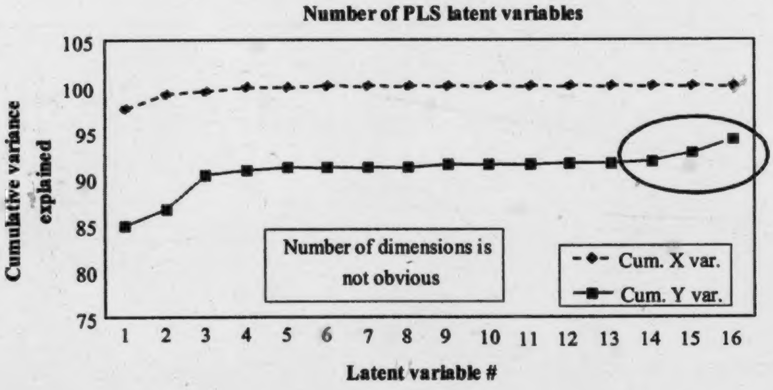


Figure 2.14: Plot of the cumulative X and Y variances captured by successive PLS dimensions

singular values are shown in Table 2.2. It is clear from Table 2.2 that the singular values of the residual space are close to unity and hence there is no ambiguity in the number of underlying components in the signal space. On the other hand, the plot showing the cumulative variance captured by successive dimensions in PLS does not give an unequivocal result for deciding the number of underlying signal components (Fig. 2.14).

The results of the predictions are shown for the validation data sets in Figs. 2.15 and 2.16. The performance indices for the tailings and flotation tailings soft sensor are given

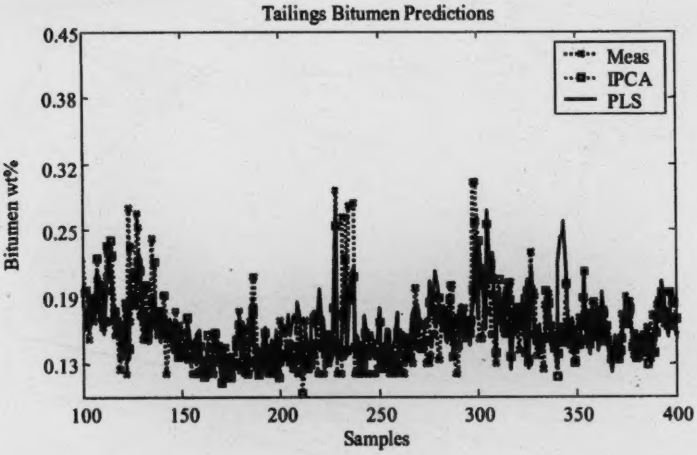


Figure 2.15: Predictions of Tailings Bitumen using PLS and IPCA techniques

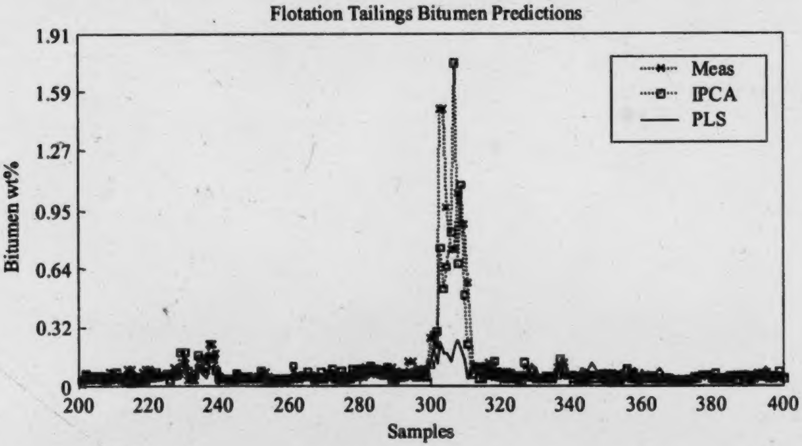


Figure 2.16: Predictions of Flotation Tailings Bitumen using PLS and IPCA techniques

in Table 2.3. The froth bitumen soft-sensor is based on PLS regression (Fig. 2.17). Poor sampling intervals prevent us from getting good predictions for this variable. However, we are still able to provide a good monitoring tool using the other two soft sensors as shown in Fig. 2.18. From the predictions, it is clear that there is great potential for the use of soft sensors for predicting bitumen recovery.

The soft sensors have been implemented online in Suncor Extraction's Distributed Control System (DCS) since June 2002, and their Plant historian (Fig. 2.18). The soft-

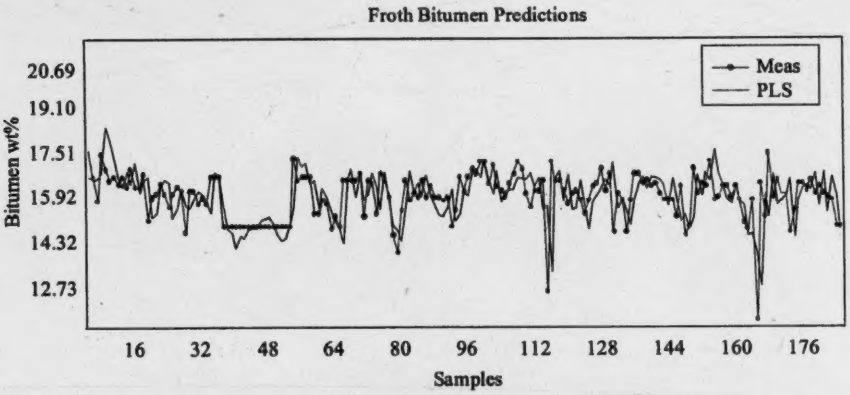


Figure 2.17: Predictions of Froth Bitumen using PLS Regression

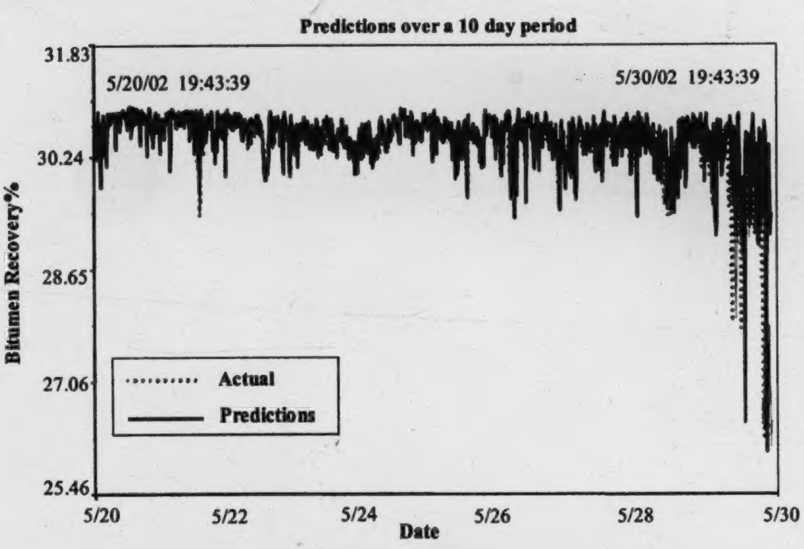


Figure 2.18: Online prediction of Bitumen Recovery

sensor based monitoring scheme has been functional for the last two years and the results are encouraging. These predictions are being used for monitoring the bitumen recovery in the separation cell. This simple tool, which gives an advance warning of an impending drop in the efficiency has helped to improve the operation.

2.4 Steady-state Fault Detection and Isolation (FDI)

When the process remains close to a particular steady-state simple static models identified using multivariate statistical techniques on routine operating data can be used for process monitoring (Kresta *et al.* 1991, Wise and Gallagher 1996). While these models and the monitoring methods developed using these models are useful for fault detection, they are unreliable for fault isolation. In particular, the contribution plot approach for fault diagnosis in PCA-based process monitoring has to be supplemented by "cause-and-effect" reasoning from the user. Nevertheless, these methods are useful because of their simplicity, data-crunching ability and opportunity for involving operating personnel as a part of the monitoring scheme.

2.4.1 FDI using PCA and contribution plots

Model identification using PCA has been described in Sec. 2.2. Process monitoring in the PCA framework is usually performed by monitoring the T^2 and SPE statistics. Following Eq. 2.9 the SVD of the data matrix **Z** yields scores and loadings defined as follows:

$$\begin{aligned} \mathbf{Z} &\approx \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^T \\ \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^T &= \mathbf{T}_s\mathbf{P}_s^T \end{aligned} \tag{2.29}$$

where, $\mathbf{T}_s = \mathbf{U}_1\mathbf{S}_1$ are the scores and $\mathbf{P}_s = \mathbf{V}_1$ are the loadings. Clearly, the scores are linear combinations of the variables in the **Z** matrix. They are also known as the *principal components*.

$$\mathbf{Z}\mathbf{P}_s = \mathbf{T}_s \tag{2.30}$$

For purposes of monitoring, PCA can be assumed to achieve a decomposition of the data matrix $\mathbf{Z} = \mathbf{Z}_s + \mathbf{Z}_r$ into a signal portion, $\mathbf{Z}_s = \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^T = \mathbf{T}_s\mathbf{P}_s^T$ and a residual portion $\mathbf{Z}_r = \mathbf{U}_2\mathbf{S}_2\mathbf{V}_2^T$. The signal space is n_s dimensional and the residual space is n_r dimensional, *i.e.*, $\mathbf{Z}_s \in \Re^{N\times n_s}$ and $\mathbf{Z}_r \in \Re^{N\times n_r}$. Following the notation used in Sec. 2.2, $n_r = r$ and $n_s = m + p - r$.

Under the assumption that the underlying signals in the **Z** matrix are independent and identically distributed with Gaussian densities, the T^2 statistic can be formulated to monitor the components in the signal space.

The T^2 statistic based on the first n_s PCs is defined as

$$T^2(t) = \sum_{i=1}^{n_s} \mathbf{T}_s(t,i)^2 \tag{2.31}$$

where $\mathbf{T}_s(t,i)$ refers to the t^{th} row and i^{th} column in the $\mathbf{T}_s$ matrix. Confidence limits for T^2 statistic at a level of $(1-\alpha)$ can be derived by noticing that this statistic has an F-distribution.

$$T^2_{N,n_s} = \frac{(N-1)n_s}{N-n_s} F_{n_s,N-n_s} \tag{2.32}$$

where $F_{n_s,N-n_s}$ is the upper $100\alpha\%$ critical point of the F distribution with n_s and $(N-n_s)$ degrees of freedom.

Similarly, the Squared Prediction Error (SPE) is used to monitor the residual space. Let $\mathbf{z}(t) \in \Re^{1\times(m+p)}$ denote the t^{th} observation. The r-dimensional residual vector is given by $\mathbf{r}(t) = \mathbf{z}(t)\mathbf{V}_2$, where $\mathbf{V}_2$ is defined in Eq. 2.9. The SPE is then defined as:

$$SPE(t) = \mathbf{r}(t)\mathbf{r}(t)^T \tag{2.33}$$

The confidence limits for SPE are given by Jackson and Mudholkar (1979). This test suggests the existence of an abnormal condition when $SPE > Q_\alpha$, where Q_α is defined as:

$$Q_\alpha = \Theta_1 \left[1 + \frac{c_\alpha h_0 \sqrt{2\Theta_2}}{\Theta_1} + \frac{\Theta_2 h_0 (h_0 - 1)}{\Theta_1^2} \right]^{\frac{1}{h_0}} \tag{2.34}$$

where,

$$\Theta_i = \sum_{j=n_s+1}^{m+p} (s_j^2)^i; \quad for\ i = 1,2,3 \tag{2.35}$$

and

$$h_0 = 1 - \frac{2\Theta_1\Theta_3}{3\Theta_2^2} \tag{2.36}$$

c_α is the confidence limit for the $1-\alpha$ percentile in a normal distribution and s_j refers to the singular value corresponding to the j^{th} component. Fault diagnosis is performed in the PCA framework using SPE contribution plots. Notice that the residual $\mathbf{r}(t)$ can be decomposed as follows:

$$\begin{aligned} \mathbf{r}(t) = \mathbf{z}(t)\mathbf{V}_2 &= \left[\begin{array}{cccc} z_1(t) & z_2(t) & \cdots & z_{m+p}(t) \end{array}\right] \left[\begin{array}{c} \mathbf{V}_2(1,:) \in \Re^{1\times r} \\ \vdots \\ \mathbf{V}_2(m+p,:) \in \Re^{1\times r} \end{array}\right] \\ &= z_1(t)\mathbf{V}_2(1,:) + \cdots + z_{m+p}(t)\mathbf{V}_2(m+p,:) \\ &= \mathbf{r}_1(t) + \cdots + \mathbf{r}_{m+p}(t) \end{aligned} \tag{2.37}$$

where, $\mathbf{V}_2(i,:)$ refers to the i^{th} row of the $\mathbf{V}_2$ matrix. $\mathbf{r}_i(t)$ can be considered as the contribution of the i^{th} variable to the residual. Hence the fractional contribution of the

i^{th} variable to the SPE is:

$$C_i(t) = \frac{\mathbf{r}_i(t)\mathbf{r}_i(t)^T}{\mathbf{r}(t)\mathbf{r}(t)^T} \tag{2.38}$$

2.5 Turbine Failure Diagnosis

In this application, we present the results of PCA and contribution plots for failure diagnosis in a unit of a coal-fired power plant. The capacity of the unit was around $500MW$ and a safety-trip occurred in this unit. Following the trip it was discovered that the turbine bearings were damaged. This resulted in nearly two weeks down-time and consequently, a significant loss of production. Our objectives of the failure diagnosis were two-fold:

1. Diagnose the reasons for the trip that occurred in the plant.
2. Find a probable cause for fatigue in the bearings of the turbine.

The main reasons for choosing PCA for the diagnosis was the requirement of dimensionality reduction and the existence of physical sensor redundancies which made the covariance matrix ill-conditioned. The unit under investigation has $12,897$ measured variables. The analysis was based on one week of data sampled once every 15 seconds. Reducing the sampling frequency by $\frac{1}{4}^{th}$ resulted in $10,741$ values for each of these tags. Using process knowledge and averaging, the number of variables was reduced to 97. On performing PCA the number of significant dimensions turned out to be 8. This is a large reduction in the dimensionality of the problem.

2.5.1 Immediate reasons for the trip

Using the SPE chart (Fig. 2.19) it was possible to predict the occurrence of the trip with a lead time of at least 80 minutes.

Possible reasons for the deviation of the SPE can be found from the contribution plot which is the plot of the individual variable contributions (Eq. 2.38) to the SPE. Since the fault occurred at the sample 10551, the immediate reasons for the deviation are found by looking at samples before this. At sample 10550 the contribution plot (Fig. 2.20) shows that the main reason for the deviation is an oil pressure. As noted earlier, diagnosis from the contribution plot can be misleading. Hence it is necessary to confirm the diagnosis. To improve the confidence in the diagnosis the trend plot of the percentage contribution of the oil pressure tag to the SPE and the absolute contribution of the oil pressure tag to the SPE are shown in Fig. 2.21 and Fig. 2.22.

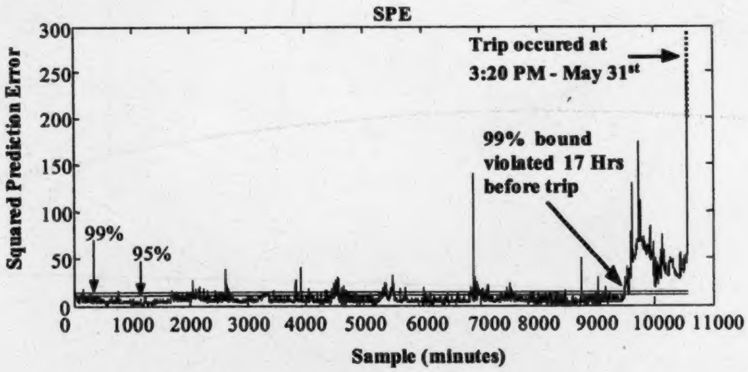


Figure 2.19: Fault detection using the Squared Prediction Error

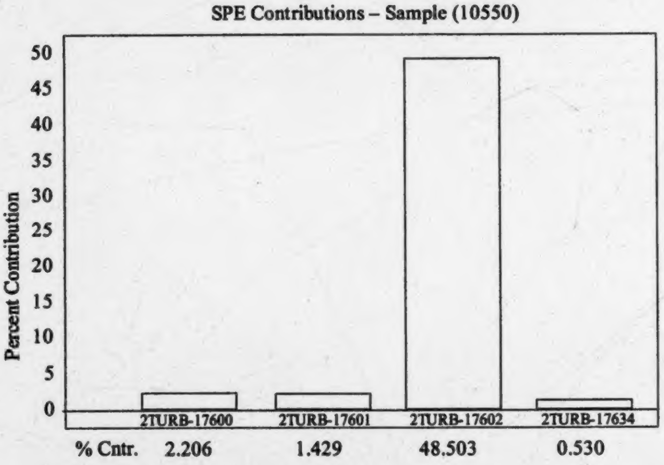


Figure 2.20: Contribution plot at sample 10551

Following these results, a look at the trend plot of the variable in question shows an abnormal deviation in this variable (Fig. 2.23).

2.5.2 Diagnosis of Bearing Fatigue

A look at the zoomed-in SPE chart (Fig. 2.24) shows that it starts deviating from sample 9476.

Eccentricity: This abnormal deviation is related to the eccentricity of the turbine shaft and an increase in the bearing temperatures. The variable *Eccentricity* plays an important role

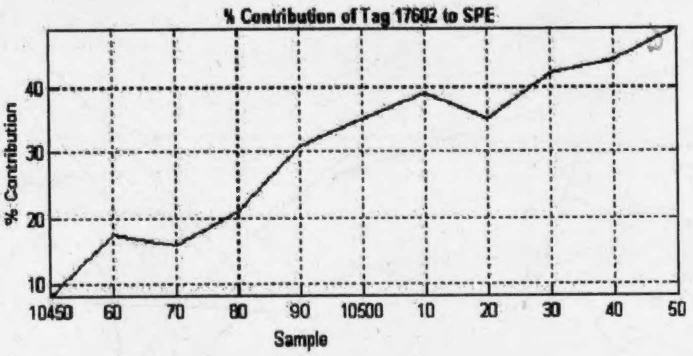


Figure 2.21: Percentage contribution of oil pressure to SPE

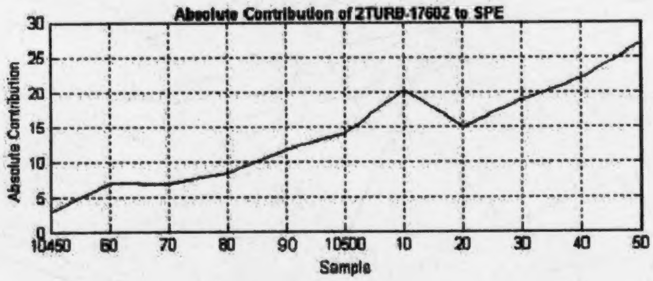


Figure 2.22: Absolute contribution of oil pressure to SPE

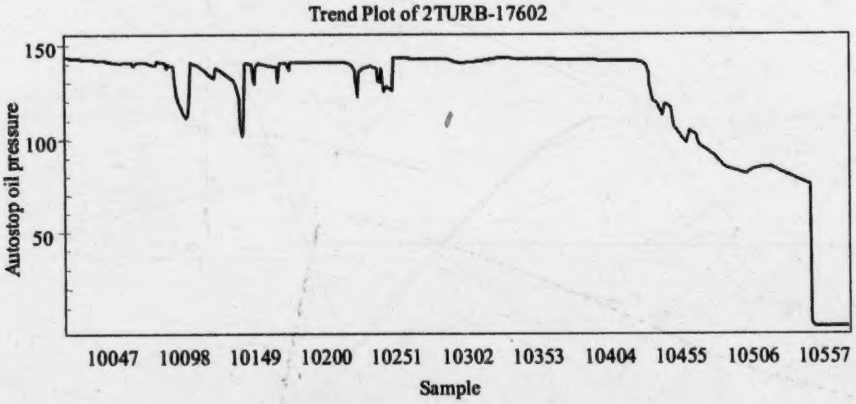


Figure 2.23: Trend plot of oil pressure

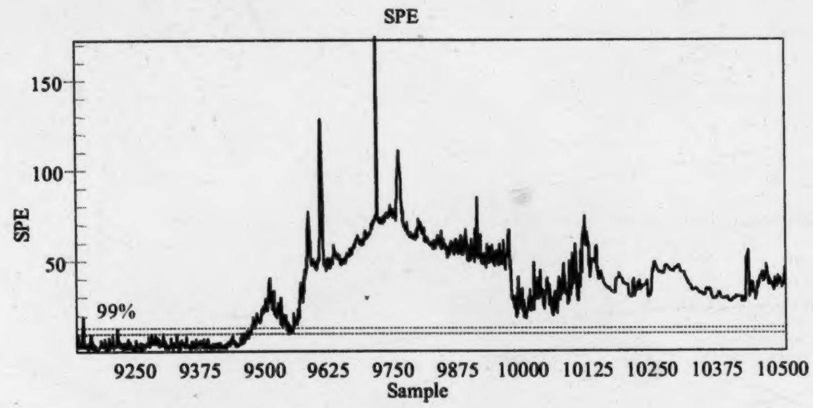


Figure 2.24: Deviation in SPE - Initial stages

in indicating a measure of the *wear* of the bearings in the turbine. Eccentricity is a measure of the deviation of the shaft from its horizontal position (Fig. 2.25).

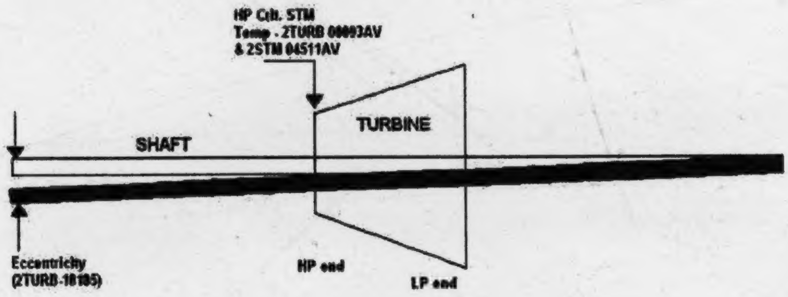


Figure 2.25: Definition of Eccentricity

Steam enters the turbine from the HP (High Pressure) end at a high pressure and temperature and leaves at the LP (Low Pressure) end at a lower pressure and temperature. Hence there is more relative expansion of the shaft on the high temperature side than on the low temperature side. This is the cause of eccentricity. To take care of this relative expansion, the shaft is cold-aligned with an extra tension which depends on the temperature of the steam entering the turbine under normal operating conditions. Under normal operating conditions, the eccentricity is at its lowest.

Turbine Bearings: There are two types of Bearings in the Turbine - the Thrust Bearing and the Journal Bearings. Since there is a difference in pressure between the HP end and the LP end of the Turbine, the Impeller-Shaft assembly tends to shift towards the LP end. To prevent this, there is a collar in the turbine casing and one in the shaft. This arrangement is called the Thrust Bearing. The Front Face (FF) and Rear Face (RF) are named according to the convention depicted in Fig. 2.26. There are two sets of Journal Bearings, one on the Thrust side (the HP side) and on the Coupling side (the LP side), for vertical support of the shaft, when the system is at rest. The bearing assembly is also used to supply lubricating oil. The cross-section of the Journal Bearing is also shown. There is a soft material called the *babbit* on which the shaft is supported when the system is at rest. When the shaft is being rotated, the film of oil lubricates the system. When the eccentricity of the shaft increases, the friction in the bearings increases and hence the bearing temperatures also increase. Further, this leads to a gradual wearing down of the bearings.

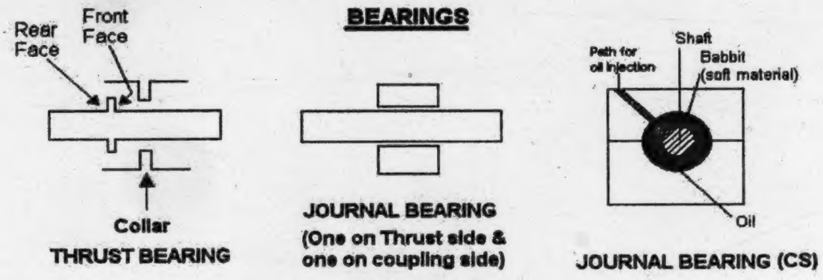


Figure 2.26: Turbine Bearings

When the system operates at a temperature lower than the designed range, the eccentricity increases, causing friction in the bearings which leads to higher bearing temperatures. There are two ways in which the operation of the system changes. One is by a change in the steam flow and the other is by a change in the steam temperature. It is seen that in this plant, the steam flow changes with the load which indicates that it is the main manipulation of some control mechanism applied to control the load. The steam temperature changes very rarely. But when this happens, the eccentricity increases. This is seen in the comparative plot of the bearing temperatures, eccentricity and the steam temperature (Fig. 2.27). It is clear that the dip in the steam temperature by about $50°F$ caused the Eccentricity to increase and move out of the normal operating range. This in turn causes fatigue in the bearings, increases the turbine bearing temperatures and causes the bearing to fail in the long run.

In conclusion, the wearing out of the bearings which is shown by the increase in the temperature of the bearings seems to be due to the increase in the Eccentricity, which in

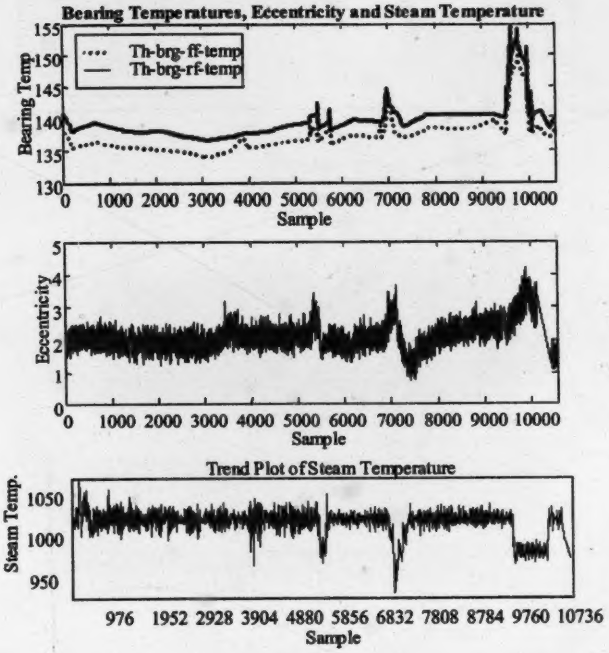


Figure 2.27: Trend plots of Bearing temperatures, Eccentricity and Steam temperature

turn is related to the decrease in the steam temperature.

The results of this application demonstrate the applicability of PCA and contribution plot based fault diagnosis. These results also point to the fact that these methods require a significant amount of process knowledge for diagnosis.

2.6 Isolation enhancements using structured residuals

It is important to note that the contribution plots discussed in Sec. 2.4 do not provide reliable diagnosis of the location of the fault (Gertler *et al.* 1999). They only indicate the variables that are most correlated with the fault. More reliable diagnosis is provided using the structured residual approach (Gertler and Singer 1990) which has been extended to deal with model identified using PCA in the isolation enhanced PCA approach (Gertler *et al.* 1999).

As stated in Sec. 2.4, the r-dimensional residual vector is given by $\mathbf{r}(t) = \mathbf{z}(t)\Theta$, where, Θ is the steady-state model obtained from first-principles or using a data-based approach like PCA. Let us consider the transposed form ($\mathbf{o}(t) = \mathbf{r}(t)^T = \Theta^T\mathbf{z}(t)^T = \mathbf{A}\mathbf{z}(t)^T$) of this residual vector for convenience. This vector is known as the Primary Residual Vector (PRV) and it can be used for fault detection. A fault detection index ($\eta_d(t)$) is constructed using the PRV, as follows:

$$\eta_d(t) = \mathbf{o}(t)^T\mathbf{Q}_o^{-1}\mathbf{o}(t) \tag{2.39}$$

where, $\mathbf{Q}_o$ is the covariance matrix of the primary residual vector calculated from a training data set collected under normal, fault-free, conditions. The fault detection index is a random variable with a χ^2 distribution with r degrees of freedom under fault-free conditions (Romagnoli and Stephanopoulos 1981). Hence fault detection can be performed by simply comparing the value of $\eta_d(t)$ with the threshold limit obtained from a central chi-squared distribution at an appropriate confidence level, say 99%.

Fault isolation is performed by transforming the PRV into a set of scalar structured residuals $\mathbf{s}(t)$ by multiplying it by a transformation matrix $\mathbf{W}$. Each of these structured residuals is designed to be insensitive to a specified subset of faults while being sensitive to the remaining faults. For example, it is possible to design the $\mathbf{W}$ matrix so that each element of $\mathbf{s}(t)$ is insensitive to one particular fault while being sensitive to other faults. The scalar structured residuals can then be used to construct fault isolation indices.

$$\eta_{i,j}(t) = \frac{s_j^2(t)}{q_{sj}} \quad j = 1, \cdots, k \tag{2.40}$$

where q_{sj} is the variance of the j^{th} structured residual calculated from a training data set collected under normal, fault-free, conditions. These isolation indices are χ^2 distributed random variables with one degree of freedom. The fault codes of this formulation are summarized in an incidence matrix such as the one shown in Table 2.4. In this table, a "0" indicates the insensitivity of a structured residual to a fault, while a "1" indicates sensitivity to the fault.

2.6.1 Maximized sensitivity enhancements

While the structured residual approach provides a framework for performing fault detection and isolation, the choice of the structured residual transformation matrix $\mathbf{W}$ is arbitrary. In addition, it is possible to design this matrix in a way such that the structured residuals have maximized sensitivity to a subset of faults while being insensitive to a particular subset of faults (Qin and Li 1999). This leads to the Structured Residual Approach with Maximized Sensitivity (SRAMS) design criterion:

Choose the rows $\mathbf{w}_i$ *of the transformation matrix* $\mathbf{W}$ *such that* $\mathbf{s}_i(t)$ *is insensitive to the* i*th fault but most sensitive to the others (Qin and Li 1999).*

Table 2.4: *Incidence matrix to characterize the isolation logic*

	f_1	f_2	f_3	$\dots$	f_k
$s_1(t)$	1	0	0	$\dots$	0
$s_2(t)$	0	1	0	$\dots$	0
$s_3(t)$	0	0	1	$\ddots$	0
$s_4(t)$	0	0	0	$\ddots$	0
$\vdots$	$\vdots$	$\ddots$	$\ddots$	$\ddots$	$\vdots$
$s_k(t)$	0	$\dots$	$\dots$	$\dots$	1

Mathematically, this leads to

$$\mathbf{w}_i = \arg\max_{\mathbf{w}_i} \sum_{j \neq i} \frac{\left(\mathbf{w}_i^T \mathbf{a}_j\right)^2}{\|\mathbf{w}_i\|^2 \|\mathbf{a}_j\|^2} \tag{2.41}$$

subject to, $\mathbf{w}_i^T \mathbf{a}_i = 0$, where, $\mathbf{a}_j$ refers to the j^{th} column of the A matrix.

The scalar structured residual and SRAMS formulations are now illustrated by a flow network example.

Flow network example

Consider the hypothetical flow network shown in Fig. 2.28. This flow network has 8

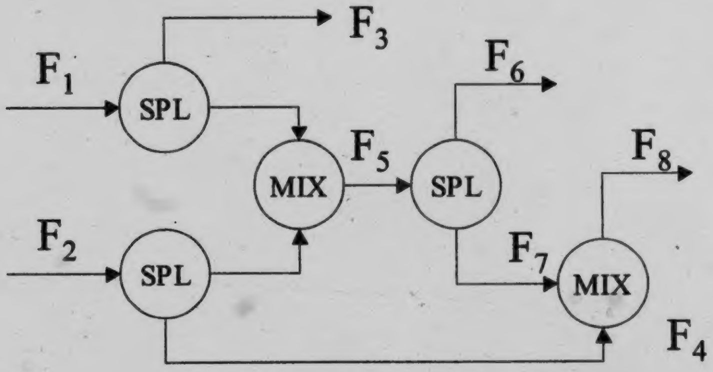


Figure 2.28: Flow network with 8 measurements and 3 constraints

measured flows and 3 constraints. The constraining equations can be summarized as:

$$\begin{aligned} F_1 + F_2 &= F_3 + F_4 + F_5 \\ F_5 &= F_6 + F_7 \\ F_8 &= F_4 + F_7 \end{aligned} \tag{2.42}$$

Assume that, $\mathbf{F}(t) = [F_1 \;\; F_2 \;\; \cdots \;\; F_8]^T$. In the absence of measurement noise and faults the constraints can be expressed as, $\mathbf{AF}(t) = \mathbf{0} \in \Re^4$, where,

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & -1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 1 \end{bmatrix} \tag{2.43}$$

To investigate fault directions, let us consider fault directions orthogonal to the fault direction of F_8, $[0\ 0\ 1]^T$. Notice immediately, that it is not possible to distinguish a sensor fault in F_8 from some other process related event, like an accumulation in the mixing device immediately preceding this variable, unless additional information is given regarding the difference in time trajectories of these events. When a fault Δf_8 occurs the primary residual vector is $\mathbf{o} = [0 \;\; 0 \;\; 1]^T f_8$, *i.e.*, it is of magnitude f_8 along the direction indicated by the column in the A-matrix corresponding to the variable F_8. The row vector $\mathbf{w}_8$ which is used to obtain the structured residual insensitive to a fault in F_8 should obviously be of the form $[w_8(1) \;\; w_8(2) \;\; 0]$. The difference between the structured residual approach and the SRAMS approach is in the way $w_8(1)$ and $w_8(2)$ are chosen. This is illustrated in Fig. 2.29, where the fault directions are indicated in the 2-D residual subspace which is orthogonal to faults in F_8. While, the ordinary structured residual approach can choose the $\mathbf{w}_8$ vector in an arbitrary direction, the maximized sensitivity approach chooses it in a direction which minimizes the angle made to all the fault directions. Hence, while the ordinary structured residual vector may inadvertently choose $\mathbf{w}_8$ orthogonal to one of the fault directions, such an event cannot happen inadvertently with the maximized sensitivity approach.

Notice however, that the SRAMS approach can also fail. For example, consider the situation where F_5 is not measured. In this case, the SRAMS approach chooses $\mathbf{w}_8$ as $[0 \quad -1 \quad 0]$. Hence the structured residual, while being orthogonal to F_8, has also become orthogonal to F_1, F_2, F_3 and F_4. The problem is that the entire residual space orthogonal to F_8 has not been captured. Only the principal direction has been captured. In order to rectify this deficiency, the Structured Residual Vector (SRV) approach with maximized sensitivity (Li and Shah 2002) constructs a 2-D structured residual vector for this problem to characterize the entire residual space orthogonal to F_8. It is superior to the scalar structured residual approaches in terms of sensitivity. This is illustrated by the following simulated case study involving the steady-state FDI in a quadruple tank process.

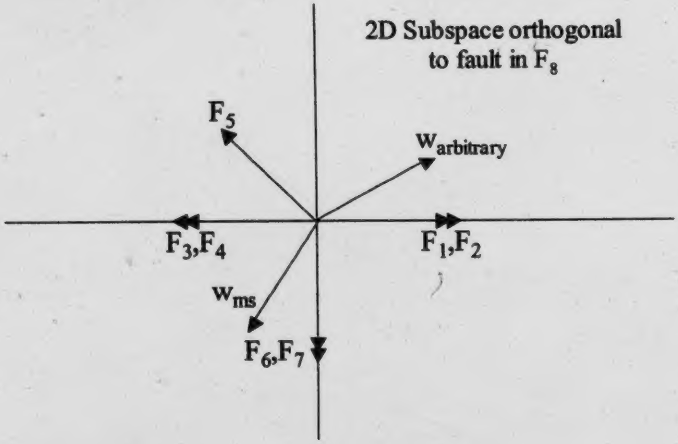


Figure 2.29: Fault directions orthogonal to fault in F_8

2.7 FDI in a Quadruple tank process

In this section, the results of Monte Carlo simulations investigating the effect of residual generation and decision rule parameters on the steady-state flow network model of the quadruple tank process (Johansson 2000) are presented. The objective of these simulations was to study the effect of various parameters on the performance of FDI algorithms. By choosing these parameters appropriately, desired sensitivity is obtained in the FDI algorithm. A schematic diagram of the quadruple tank process is shown in Fig. 2.30. A steady-state flow network based on this process is shown in Fig 2.31.

We assume that eight flow rates are measured in this process (F_1 - F_8). The mass balance can be summarized by the following equations:

$$\begin{aligned} F_5 &= F_2 + F_3 \\ F_6 &= F_1 + F_4 \\ F_7 &= F_1 + F_3 \\ F_8 &= F_2 + F_4 \end{aligned} \tag{2.44}$$

Consequently, we can summarize,

$$\begin{bmatrix} 0 & 1 & 1 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_8 \end{bmatrix} = \mathbf{0} \Rightarrow \mathbf{AF} = \mathbf{0} \tag{2.45}$$

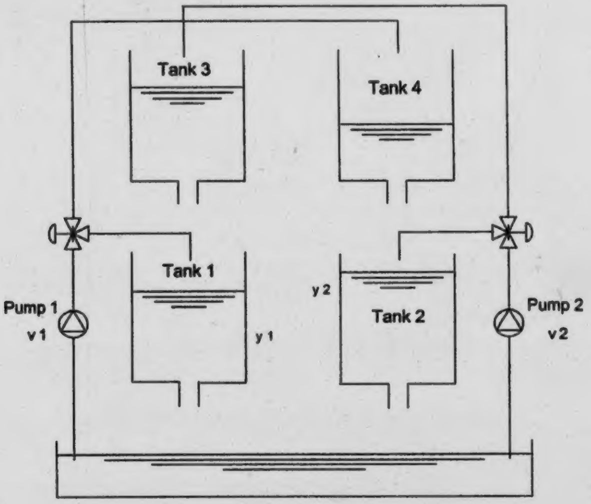

Tank 3
Tank 4
Tank 1
Tank 2
Pump 1
v1
y1
y2
Pump 2
v2

Figure 2.30: Schematic of the four water tank system

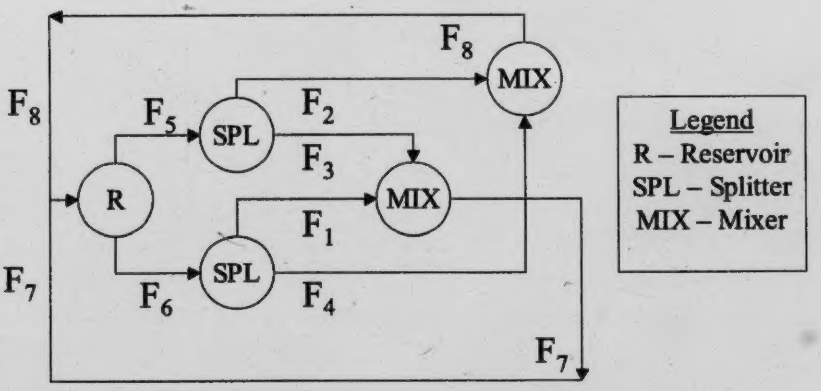

F_8
F_8
MIX
F_5
SPL
F_2
F_3
R
MIX
F_1
SPL
F_7
F_6
F_4
F_7
Legend
R – Reservoir
SPL – Splitter
MIX – Mixer

Figure 2.31: Steady-state flow network for quadruple tank process

Data sets corresponding to 100 different noise realizations were generated, with 1000 samples in each data set. This corresponds to $100 \times 1000 = 100,000$ samples for performing FDI. In addition to the no-fault case, faults of sizes (0.5, 1, 2, 3 and 5σ) were introduced in each of the 8 sensors, where σ is the standard deviation of the noise in the sensor. This results in 8 different types of data sets with 5 different fault sizes and 1 non-faulty data set. Each of these cases has 100,000 samples. To make things simple, only the single fault hypothesis was considered. The performance of 6 different FDI settings

Table 2.5: *Different FDI settings for montecarlo experiments*

Case	Model	Isolation Strategy	Decision Logic
1	1st principles	Scalar Structured Residual (SRS)	Multiple testing
2	1st principles	SRAMS	Multiple testing
3	1st principles	Structured Residual Vector (SRV)	Multiple testing
4	1st principles	Scalar Structured Residual (SRS)	Single testing
5	1st principles	SRAMS	Single testing
6	1st principles	Structured Residual Vector (SRV)	Single testing

corresponding to different choices of the residual generation approach and decision rule was studied. These choices are shown in Table 2.5.

In all cases, the first principles model shown in Eq. 2.45 is used. This model is used to generate the primary residual vector $\mathbf{o}(t)$ used for fault detection. Once the primary residual vector is generated, the structured residual is formed for fault isolation.

In the scalar structured residual method, the transformation matrix used to obtain the structured residual is,

$$\mathbf{W} = \begin{bmatrix} 2 & 1 & -1 & 1 \\ 1 & 2 & 1 & -1 \\ 1 & 2 & -1 & 1 \\ 2 & 1 & 1 & -1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \tag{2.46}$$

This results in 8 scalar structured residuals for performing fault isolation.

In the SRAMS approach, the transformation matrix used to obtain the structured residual is shown in Eq. 2.47. In this case as well, 8 scalar structured residuals are obtained and fault isolation was performed using these.

$$\mathbf{W}_{ms} = \begin{bmatrix} -0.71 & 0 & 0 & -0.71 \\ 0 & -0.71 & -0.71 & 0 \\ 0 & -0.71 & 0 & -0.71 \\ -0.71 & 0 & -0.71 & 0 \\ 0 & -0.68 & -0.52 & -0.52 \\ -0.68 & 0 & -0.52 & -0.52 \\ -0.52 & -0.52 & 0 & -0.68 \\ -0.52 & -0.52 & -0.68 & 0 \end{bmatrix} \tag{2.47}$$

In the Structured Residual Vector approach, structured residual vectors, $\mathbf{s}_i(t) \in \Re^3$, $\forall\ i = 1, \cdots, 8$ were generated using the transformation matrices shown in Eqs. 2.48 to 2.55.

$$\mathbf{W}_1 = \begin{bmatrix} -0.71 & 0 & 0 & -0.71 \\ 0.50 & -0.50 & 0.50 & -0.50 \\ 0.50 & 0.50 & -0.50 & -0.50 \end{bmatrix} \tag{2.48}$$

$$\mathbf{W}_2 = \begin{bmatrix} 0 & -0.71 & -0.71 & 0 \\ -0.50 & 0.50 & -0.50 & 0.50 \\ 0.50 & 0.50 & -0.50 & -0.50 \end{bmatrix} \tag{2.49}$$

$$\mathbf{W}_3 = \begin{bmatrix} 0 & -0.71 & 0 & -0.71 \\ -0.50 & 0.50 & 0.50 & -0.50 \\ 0.50 & 0.50 & -0.50 & -0.50 \end{bmatrix} \tag{2.50}$$

$$\mathbf{W}_4 = \begin{bmatrix} -0.71 & 0 & -0.71 & 0 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.50 & 0.50 & -0.50 & -0.50 \end{bmatrix} \tag{2.51}$$

$$\mathbf{W}_5 = \begin{bmatrix} 0 & -0.68 & -0.52 & -0.52 \\ 0 & 0 & -0.71 & 0.71 \\ 0 & 0.74 & -0.48 & -0.48 \end{bmatrix} \tag{2.52}$$

$$\mathbf{W}_6 = \begin{bmatrix} -0.68 & 0 & -0.52 & -0.52 \\ 0 & 0 & -0.71 & 0.71 \\ 0.74 & 0 & -0.48 & -0.48 \end{bmatrix} \tag{2.53}$$

$$\mathbf{W}_7 = \begin{bmatrix} -0.52 & -0.52 & 0 & -0.68 \\ 0.71 & -0.71 & 0 & 0 \\ -0.48 & -0.48 & 0 & 0.74 \end{bmatrix} \tag{2.54}$$

$$\mathbf{W}_8 = \begin{bmatrix} -0.52 & -0.52 & -0.68 & 0 \\ 0.71 & -0.71 & 0 & 0 \\ -0.48 & -0.48 & 0.74 & 0 \end{bmatrix} \tag{2.55}$$

Once the structured residuals are formed, the incidence matrix used to characterize the fault codes is given by Table 2.6. As far as the decision making is concerned, there are two possibilities, multiple testing (Cases 1 to 3) and single testing (Cases 4 to 6). To illustrate these testing strategies, let us consider the fault f_1. In the case of multiple testing, the test which has to be satisfied is $\eta_{i,1} < \chi_3^2(\alpha)$ and $\eta_{i,j} > \chi_3^2(\alpha), \quad \forall \;\; j = 2, \cdots, 8$. If all the isolation indices are greater than the limits, the fault detection test has been falsified. If more than one isolation index is less than the limit, it corresponds to an uncertain fault, *i.e.*, the fault code is not present in this incidence matrix. In practice, this leads to a large number of uncertain results. In order to avoid these problems, the single testing strategy is proposed. In this strategy, we only have to consider the minimum among the fault isolation indices. Since, the fault detection test has already established the presence of a fault, we simply assign the fault to the minimum among the fault isolation indices.

Table 2.6: *Incidence matrix to characterize the isolation logic*

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
$s_1(t)$	0	1	1	1	1	1	1	1
$s_2(t)$	1	0	1	1	1	1	1	1
$s_3(t)$	1	1	0	1	1	1	1	1
$s_4(t)$	1	1	1	0	1	1	1	1
$s_5(t)$	1	1	1	1	0	1	1	1
$s_6(t)$	1	1	1	1	1	0	1	1
$s_7(t)$	1	1	1	1	1	1	0	1
$s_8(t)$	1	1	1	1	1	1	1	0

The results of these monte-carlo simulations are summarized in the FDI performance tables (Tables A.1 to A.12) in the Appendix. The results show that,

- Fault detection and isolation tests improve with an increase in the size of the fault.
- The performance of the SRV-based approach is much better than the scalar structured residual and the SRAMS approaches.
- The SRAMS approach is very poor in isolating faults in $F_5, \cdots, F_8$.
- The multiple testing strategy gives rise to a large number of misclassifications in fault isolation.
- The SRV-based approach with single testing minimizes misclassification and uncertain results and has the best performance in fault isolation.

2.8 Conclusions

Principles of steady-state model identification and fault diagnosis have been presented and illustrated using simulations and industrial case-studies. Two industrial case-studies involving PCA and PLS have been presented, to demonstrate the application of multivariate statistical techniques for process monitoring. In addition, some of the issues involved in developing inferential sensors from archived historical data have also been addressed. A novel approach for the joint identification of steady-state models and the noise covariance matrix has been described and illustrated using an industrial case-study. The superiority of SRV-based fault isolation over scalar structured residual approaches has been demonstrated. Optimal choices for the residual generation mechanism and decision rule criteria for SRV-based FDI have been established using a simulated quadruple tank process.

3

Discrete-time dynamic model identification and fault diagnosis

3.1 Overview

A common problem faced in the process industry is that detailed mathematical models of processes are not available. Since all FDD methods require some sort of model this constraint has important consequences on the safety and performance of these processes. Among the different faults that affect processes, sensor and actuator faults affect the rest of the system in an additive way. They can be diagnosed using discrete-time input/output (I/O) models of processes since they affect the process at the input or output ports. Such I/O models can be identified using techniques such as Prediction Error Methods (PEM) or Subspace Identification Methods (SIM). Following the identification of these I/O models, sensor and actuator faults can be detected and diagnosed using parity-space or observer-based techniques or their extensions. In addition, these methods can be applied to plants operating under closed-loop conditions[1]. In the following sections we describe the application of Structured Residual Vector (SRV) approaches for the detection and diagnosis of sensor and actuator faults using simulations on a quadruple tank process and experimental studies on a pilot-scale Continuous-flow Stirred Tank Heater process.

---

[1] Sections of this chapter have been presented or published as:

1. H. Raghavan and S.L. Shah, "Diagnosis of sensor faults in closed-loop systems", Presented in AIChE Annual Conference, Indianapolis, USA, Nov, 2002.
2. H. Raghavan and S.L. Shah, "Data-driven approaches for detection and diagnosis of Sensor faults", Presented in the Western Canadian Process Control Conference, Calgary, Canada, June, 2002.

3.2 I/O faults in open-loop and closed-loop systems

In this section, we illustrate the consequences of the occurrence of actuator and sensor faults in systems operating under open-loop and closed-loop conditions using simulations of a quadruple tank process.

A schematic diagram of the process is shown in Fig. 3.1. The objective is to control the level in the lower two tanks using two pumps. The process inputs are v_1 and v_2 (input voltages to the pumps) and the outputs are y_1, y_2 (voltages from the level measurement devices) and y_3 (voltage from a measurement device providing a redundant measurement based on the difference in the levels in Tanks 1 and 2). Mass balances and Bernoulli's law yield (Johansson 2000):

$$\begin{aligned}
\frac{dh_1}{dt} &= \frac{\gamma_1 k_1}{A_1} v_1 - \frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_3}{A_1}\sqrt{2gh_3} \\
\frac{dh_2}{dt} &= \frac{\gamma_2 k_2}{A_2} v_2 - \frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_4}{A_2}\sqrt{2gh_4} \\
\frac{dh_3}{dt} &= \frac{(1-\gamma_2) k_2}{A_3} v_2 - \frac{a_3}{A_3}\sqrt{2gh_3} \\
\frac{dh_4}{dt} &= \frac{(1-\gamma_1) k_1}{A_4} v_1 - \frac{a_4}{A_4}\sqrt{2gh_4}
\end{aligned} \tag{3.1}$$

where, A_i - cross-section of tank i, a_i - cross-section of the i^{th} outlet, and h_i - water level in the i^{th} tank. The parameters $\gamma_1, \gamma_2 \in (0,1)$ are determined from how the valves are set prior to an experiment. The flow to Tank 1 is $\gamma_1 k_1 v_1$, the flow to Tank 4 is $(1-\gamma_1)k_1 v_1$ and similarly for Tank 2 and Tank 3. The acceleration due to gravity is denoted by g. The measured level signals are $k_c h_1$, $k_c h_2$, $k_c h_3$ and $k_c h_4$, where k_c is a parameter associated with the sensor gain.

The parameter values and the chosen operating point of the laboratory process are given in Table 3.1.

The introduction of actuator and sensor biases in the process is shown in Figs. 3.2 and 3.3.

An important factor to study in the comparison of different approaches is the sensitivity of the approach. In this study we have quantified the size of the biases in terms of the noise

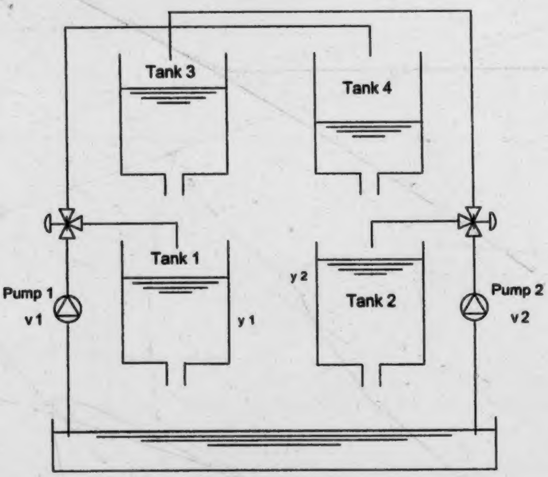

Tank 3
Tank 4
Tank 1
Tank 2
Pump 1
v1
Pump 2
v2
y1
y2

Figure 3.1: Schematic of the four water tank system

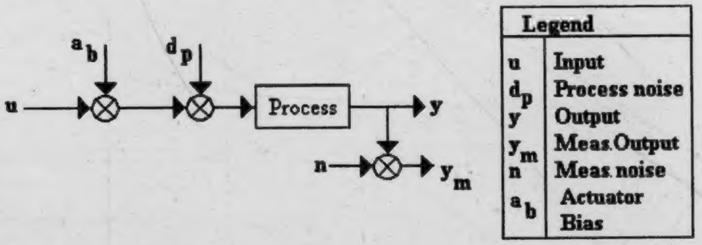

a_b
d_p
u
Process
y
n
y_m
Legend
u Input
d_p Process noise
y Output
y_m Meas.Output
n Meas. noise
a_b Actuator Bias

Figure 3.2: Schematic illustrating the introduction of actuator biases

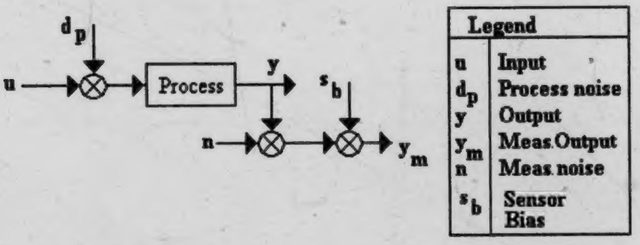

d_p
u
Process
y
s_b
n
y_m
Legend
u Input
d_p Process noise
y Output
y_m Meas.Output
n Meas. noise
s_b Sensor Bias

Figure 3.3: Schematic illustrating the introduction of sensor biases

level in the system. It is assumed that the system operates subject to

- Process noise ($\mathbf{d}_p$), which can be characterized by zero mean random Gaussian process with a standard deviation of σ ($\mathbf{d}_p \sim N(0, \sigma^2\mathbf{I}_2)$).

A_1, A_3	$[cm^2]$	28
A_2, A_4	$[cm^2]$	32
a_1, a_3	$[cm^2]$	0.071
a_2, a_4	$[cm^2]$	0.057
k_1, k_2	$[cm^3/Vs]$	3.33,3.35
h_1°, h_2°	$[cm]$	12.4,12.7
h_3°, h_4°	$[cm]$	1.8,1.4
v_1°, v_2°	$[V]$	3.00
k_c	$[V/cm]$	0.50
g	$[cm/s^2]$	981
γ_1, γ_2		0.70,0.60

Table 3.1: *Parameter values and chosen operating point of the Quadruple-tank process*

- Measurement noises ($\mathbf{n}$), which can be characterized by a zero mean random Gaussian process with a standard deviation of σ ($\mathbf{n} \sim N(0, \sigma^2 \mathbf{I}_3)$).

The value of $0.06V$ is chosen for the standard deviation σ of the noise. With this chosen value of σ, biases are introduced in the actuators and sensors at levels starting from σ. The sensitivity of different methods is compared in the sense that the size of the smallest bias, which can be detected and isolated by a particular method, is quantified.

Trend plots are plots of the variables as a function of time. In the following figures introduction of different types of faults is illustrated. The information gathered from the trend plots can be misleading under some circumstances as can be gathered from the following illustrations.

3.2.1 Open loop - actuator faults - Trend plots

When biases develop in the actuator there is a discrepancy between the intended input to the system and the actual input. This is illustrated in Fig. 3.4.

These can be important because they may lead to unforeseen changes in the outputs. It is not possible to effectively diagnose the cause of these events from trend plots or using univariate methods. This is clear from Fig. 3.5. The use of techniques like PCA for diagnosis may sometimes lead to misleading results and classify the event as a change in the operating condition caused by some unknown disturbance. However when techniques based on open-loop input-output models are used for diagnosis, it is found that the results are accurate and sensitive for detection and diagnosis of very small faults.

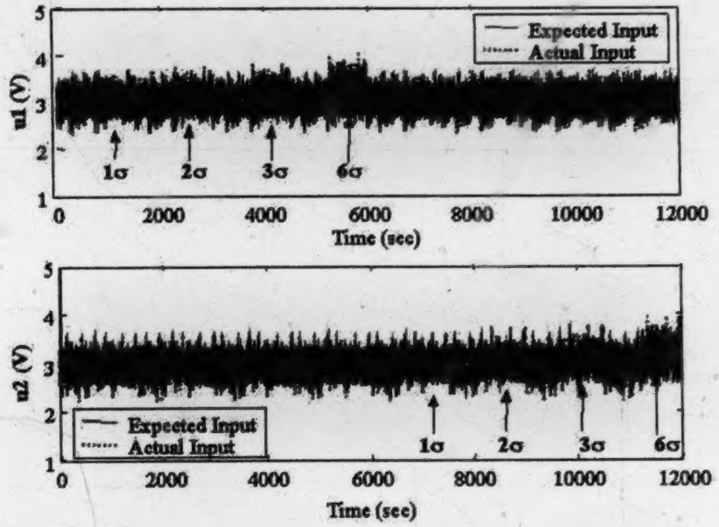


Figure 3.4: Actuator Biases: Differences between the expected inputs and true inputs

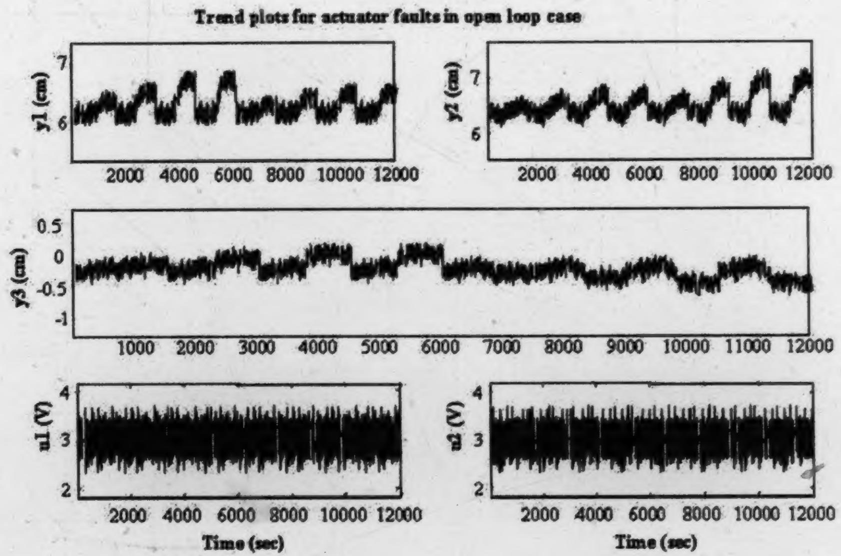

Trend plots for actuator faults in open loop case

Figure 3.5: Actuator Biases: Trend plots of the observed variables

3.2.2 Open loop - sensor faults - Trend plots

When biases develop in the sensors under open-loop conditions, there is a discrepancy between the actual state of the process and the information received about the state of the process. This is illustrated in Fig. 3.6.

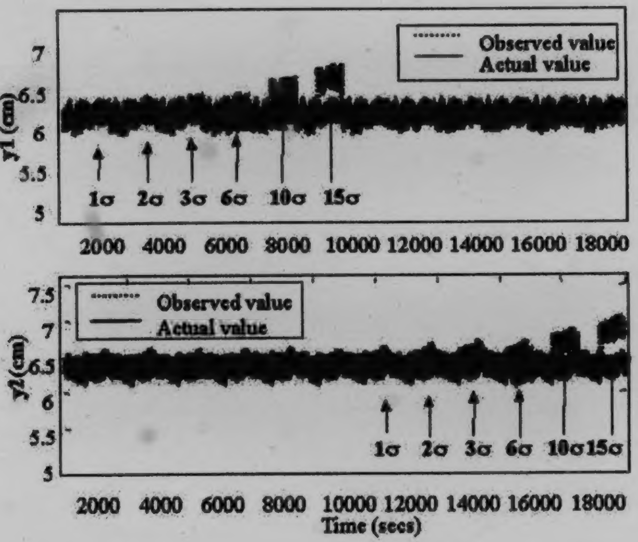


Figure 3.6: Sensor Biases: Differences between the actual state of process and observed state of the process

The trend plots of the observed variables (Fig. 3.7) show changes in the process when there are no changes in reality. The manipulations performed based on the observed variables may be incorrect and may lead to undesirable situations.

Diagnosis based on comparison of the values of redundant sensors can be effective. Diagnosis can also be performed by simple multivariate approaches like PCA or by comparing the readings of redundant sensors. In addition, Model-based FDD schemes are accurate and sensitive in diagnosing these faults.

3.2.3 Closed loop case - actuator faults - Trend plots

When small biases develop in the actuator there is a discrepancy between the intended input to the system and the actual input. This is corrected by the controller, which adjusts the input to the system so that the output remains at the desired level. Hence an actuator bias

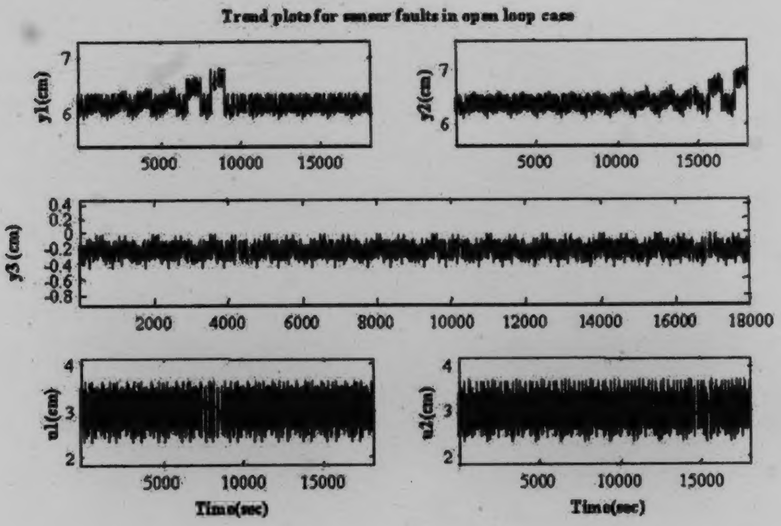


Figure 3.7: Sensor Biases: Trend plots of the observed variables

by itself is not a serious matter if the system is under closed-loop condition. It a problem only when the actuator reaches one of its constraints due to this correction.

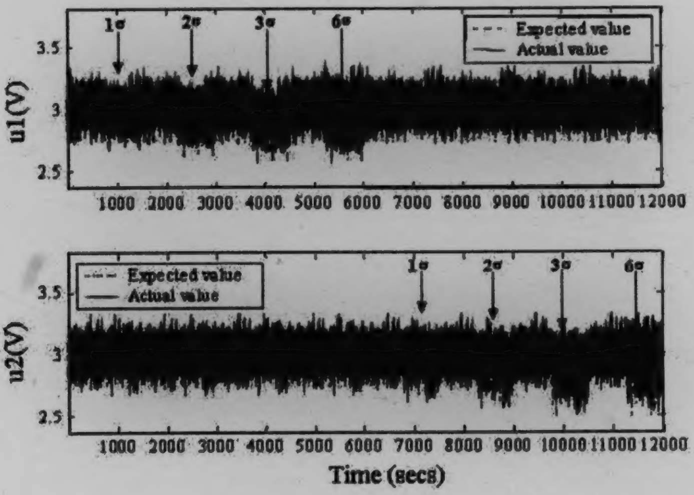


Figure 3.8: Closed-loop Actuator Biases: Differences between the expected inputs and true inputs

The fact that there is an actuator bias in a closed-loop system can be found out by just looking at the trend plots (Figs. 3.8 and 3.9) or by using univariate statistics.

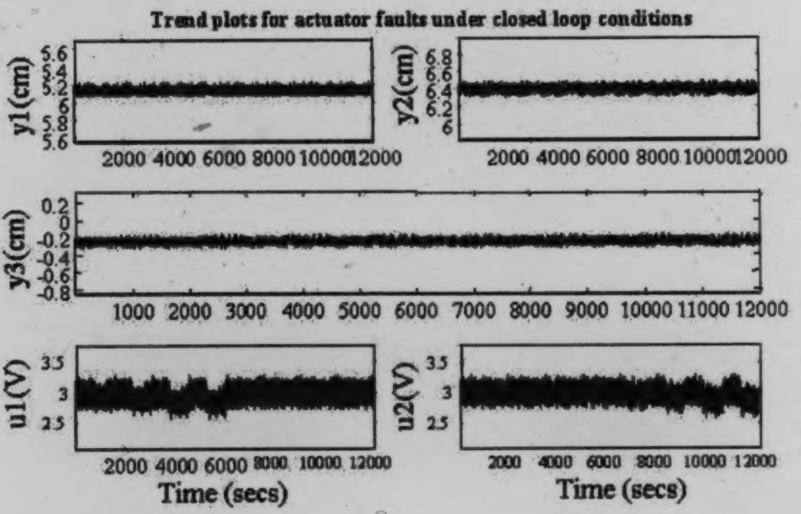


Figure 3.9: Closed-loop Actuator Biases: Trend plots of the observed variables

These faults can also be detected and accurately diagnosed by using model-based FDD techniques.

3.2.4 Closed loop case - sensor faults: Trend plots

This is by far the most serious fault among these faults. When biases develop in the sensors under closed-loop conditions, there is a discrepancy between the actual state of the process and the information received by the controller about the state of the process (Fig. 3.10). This leads the controller to perform incorrect manipulations on the process.

The controller performs incorrect manipulations on the process when sensor biases develop in the closed loop system. This is potentially dangerous because the manipulations affected by the controller may lead the process away from the desired operating point. When the process controlled is nonlinear this leads to significant deviations. The movement of the process to an undesirable operating point may have disastrous consequences. Fig. 3.11 shows the trend plots of the observed variables when biases develop in the sensors.

While diagnosis based on trend plots, univariate methods and PCA is not reliable and accurate, model-based FDD schemes are accurate and sensitive in detecting and diagnosing

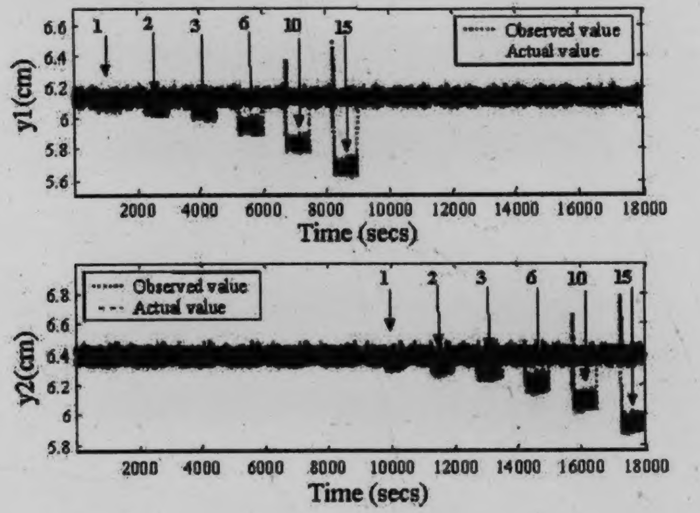


Figure 3.10: Closed-loop Sensor Biases: Differences between the observed and true outputs

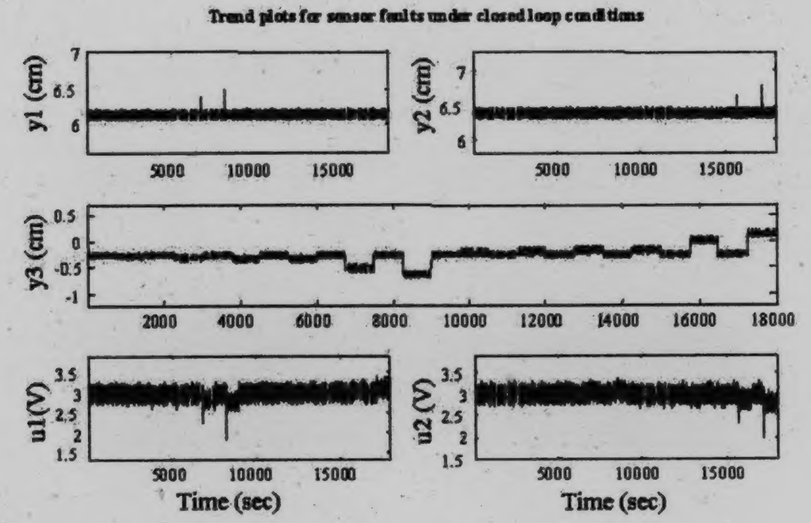


Figure 3.11: Closed-loop Sensor Biases: Trend plots of the observed variables

these faults.

3.3 Diagnosis of I/O faults using the SRV approach

The Structured Residual Vector-based (SRV-based) approach is a model-based approach that can be used for the detection and isolation of sensor and actuator faults in dynamic systems. It can be applied for FDI of processes for which state-space models are available from first principles as well as those processes for which models have to be identified from input-output data. The advantage of this approach is that fault diagnosis relevant models identified using subspace identification can be used directly for sensor and actuator FDI.

Consider a dynamic system which can be represented in the fault-free case by the following discrete-time state-space model,

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{d}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) + \mathbf{o}(k) \end{aligned} \tag{3.2}$$

where, $\mathbf{u}(k) \in \Re^l$, $\mathbf{y}(k) \in \Re^m$ are noise-free inputs and outputs respectively; $\mathbf{x}(k) \in \Re^n$ is the state vector and $\mathbf{d}(k)$ and $\mathbf{o}(k)$ are the disturbance and noise vectors respectively, which are assumed to be unmeasured multivariate white noise processes with gaussian densities and covariances $\mathbf{R}_d$ and $\mathbf{R}_o$ respectively. Further it is assumed that $\mathbf{d}(k)$ and $\mathbf{o}(k)$ are independent of the initial state $\mathbf{x}(0)$ and are mutually independent.

If the sensors or actuators are faulty, their readings will contain fault-free and fault-related values.

$$\begin{aligned} \mathbf{u}^*(k) &= \mathbf{u}(k) + \mathbf{F}_u\mathbf{f}_u(k) \\ \mathbf{y}^*(k) &= \mathbf{y}(k) + \mathbf{F}_y\mathbf{f}_y(k) \end{aligned} \tag{3.3}$$

where, $\mathbf{F}_u \in \Re^{l\times d_u}$ and $\mathbf{F}_y \in \Re^{m\times d_y}$ are matrices of fault directions and $\mathbf{f}_u \in \Re^{d_u}$ and $\mathbf{f}_y \in \Re^{d_y}$ are fault magnitude vectors. To represent a fault in the i^{th} sensor, let $\mathbf{F}_y = [0\ldots1\ldots0]^T \in Re^m$, which is the i^{th} column of the identity matrix $\mathbf{I}_m$. Similarly, to represent a fault in the i^{th} actuator, let $\mathbf{F}_u = [0\ldots1\ldots0]^T \in Re^l$, which is the i^{th} column of the identity matrix $\mathbf{I}_l$. With Eqs. 3.2 and 3.3, the problem of sensor and actuator FDI is stated as follows:

- Fault detection: indicate when $\mathbf{f}_u(k)$ and/or $\mathbf{f}_y(k)$ are non-zero; and
- Fault isolation: identify fault direction matrices $\mathbf{F}_u$ and or $\mathbf{F}_y$.

Recursion of Eq. 3.2 yields,

$$\mathbf{y}_s(k) = \mathbf{\Gamma}_s\mathbf{x}(k-s) + \mathbf{H}_s\mathbf{u}_s(k) + \mathbf{G}_s\mathbf{d}_s(k) + \mathbf{o}_s(k) \tag{3.4}$$

where,

$$\mathbf{y}_s(k) = \begin{bmatrix} \mathbf{y}(k-s) \\ \mathbf{y}(k-s+1) \\ \vdots \\ \mathbf{y}(k) \end{bmatrix} \text{ is the stacked output vector,}$$

$$\mathbf{\Gamma}_s = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}A \\ \vdots \\ \mathbf{C}A^s \end{bmatrix} \text{ is the extended observability matrix,}$$

and

$$\mathbf{H}_s = \begin{bmatrix} \mathbf{D} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{C}B & \mathbf{D} & & \vdots \\ \vdots & \ddots & \ddots & \\ \mathbf{C}A^{s-1}\mathbf{B} & \dots & \mathbf{C}B & \mathbf{D} \end{bmatrix} \quad \mathbf{G}_s = \begin{bmatrix} \mathbf{0} & \dots & & \mathbf{0} \\ \mathbf{C} & \mathbf{0} & & \vdots \\ \vdots & \ddots & \ddots & \\ \mathbf{C}A^{s-1} & \dots & \mathbf{C} & \mathbf{0} \end{bmatrix}$$

are two lower triangular block Toeplitz matrices. Note that $\mathbf{G}_s$ is completely dependent on the first ms rows of $\mathbf{\Gamma}_s$. Therefore, once $\mathbf{\Gamma}_s$ is identified, $\mathbf{G}_s$ can be derived from it. In addition, in Eq. 3.4, s is defined as the order of the parity space or the maximum detectability index of the fault (Liu and Si 1997) and is selected to be equal to n without loss of generality. The stacked vectors $\mathbf{u}_s(k)$ and $\mathbf{d}_s(k)$ are in the similar format as $\mathbf{y}_s(k)$. It is assumed that the order n is known.

Define,

$$\tilde{\mathbf{H}}_s = [\ \mathbf{I}_{ms} | \ \ -\mathbf{H}_s\ ]$$

Hence in the presence of faults,

$$\tilde{\mathbf{H}}_s \begin{bmatrix} \mathbf{y}_s(k) \\ \mathbf{u}_s(k) \end{bmatrix} = \mathbf{\Gamma}_s \mathbf{x}(k-s) + \mathbf{G}_s \mathbf{d}_s(k) + \mathbf{o}_s(k) + \tilde{\mathbf{H}}_s \mathbf{F}_{s,z} \mathbf{f}_{s,z}(k) \tag{3.5}$$

where,

$$\mathbf{F}_{s,z} = \begin{bmatrix} \mathbf{I}_{s+1} \otimes \mathbf{F}_y & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{s+1} \otimes \mathbf{F}_u \end{bmatrix} \text{ and } \mathbf{f}_{s,z}(k) = [\mathbf{f}_y^T(k-s) \dots \mathbf{f}_y^T(k) \mathbf{f}_u^T(k-s) \dots \mathbf{f}_u^T(k)]^T$$

Pre-multiplying Eq. 3.4 by a matrix $\mathbf{W}_0$, which lies in the null space of $\mathbf{\Gamma}_s$, i.e. $\mathbf{W}_0 \mathbf{\Gamma}_s = 0$, produces :

$$\boldsymbol{\varepsilon}_s(k) \equiv \mathbf{P}_s \begin{bmatrix} \mathbf{y}_s(k) \\ \mathbf{u}_s(k) \end{bmatrix} = \mathbf{W}_0 \mathbf{G}_s \mathbf{d}_s(k) + \mathbf{W}_0 \mathbf{o}_s(k) + \mathbf{W}_0 \tilde{\mathbf{H}}_s \mathbf{F}_{s,z} \mathbf{f}_{s,z}(k) \tag{3.6}$$

where $\mathbf{P}_s \equiv \mathbf{W}_0 \tilde{\mathbf{H}}_s = [\mathbf{W}_0 | - \mathbf{W}_0 \mathbf{H}_s]$ is defined as the Primary Residual Model (PRM) for fault detection. Note that $\mathbf{W}_0 \tilde{\mathbf{H}}_s$ is the fault model for the primary residual vector

(PRV) $\boldsymbol{\varepsilon}_s(k)$. Therefore, the sensor and/or actuator fault detection problem can be reduced to checking whether $\boldsymbol{\varepsilon}_s(k)$ is zero mean. $\mathbf{\Gamma}_s$ and $\mathbf{H}_s$ can be identified from training data using a subspace identification procedure and $\mathbf{W}_0$ can be calculated optimally from the identified $\mathbf{\Gamma}_s$ and $\mathbf{H}_s$ using the procedure outlined in Li and Shah (2002).

The squared weighted residual (SWR), i.e., $\eta_s(k) = e_s^T(k)\mathbf{R}_{s,e}^{-1}e_s(k)$ is used as the index for fault detection, with $e_s(k)$ being the PRV filtered using an exponentially weighted moving average (EWMA) filter and $\mathbf{R}_{s,e}$ the corresponding covariance matrix obtained from training data. $\eta_s(k)$ follows a central chi-square distribution with $(m_s - n)$ degrees of freedom, where $m_s = m \times (s+1)$. Hence the limit from a chi-squared distribution with an appropriate confidence level can be used to detect the presence of a fault in the system.

For fault isolation, the PRV is transformed into a set of Structured Residual Vectors (SRV's). In such a set, one SRV is designed to be immune to a specified subset of faults, but has maximized sensitivity with respect to the remaining faults. Hence the isolation indices follow an inverse logic. The details of the generation of the SRV are outlined in Li and Shah (2002).

3.4 Closed-loop FDD using the SRV approach

There have been many cases where model-based FDI techniques have been applied successfully in simulation studies. However there have been very few reported real-time applications even for pilot scale processes (Afonso *et al.* 1998). Many of these applications assume the availability of an accurate and detailed mathematical model of the system. In contrast many of the real-life applications require FDI under closed-loop conditions and the availability of a mathematical model from first-principles is not guaranteed. Applications carried out under such circumstances are not common in FDI literature. However, for most process systems, model-based FDI can be successfully performed under closed-loop conditions. The open-loop model can be identified through proper excitation of the process inputs. In addition, there are a number of methods available for explicitly identifying the open-loop model from closed-loop data (Forssell and Ljung 1999, Huang and Shah 1997). In the following section, we illustrate sensor FDI in a pilot-scale Continuous-flow Stirred-Tank Heater (CSTH) process under closed-loop conditions using an I/O model.

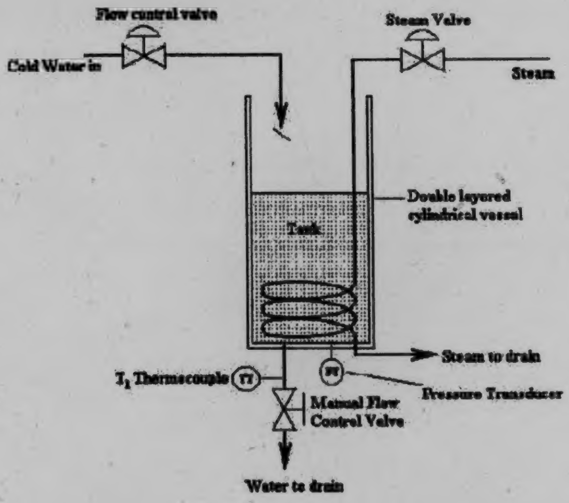

Flow control valve
Cold Water in
Steam Valve
Steam
Double layered cylindrical vessel
Tank
Steam to drain
T_1 Thermocouple
TT
PT
Pressure Transducer
Manual Flow Control Valve
Water to drain

Figure 3.12: Continuous-flow Stirred Tank Heater

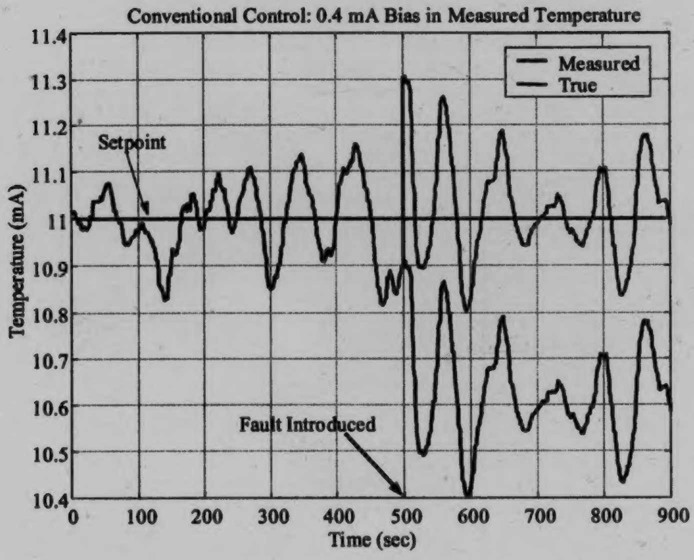


Figure 3.13: Temperature Sensor Bias under closed-loop conditions

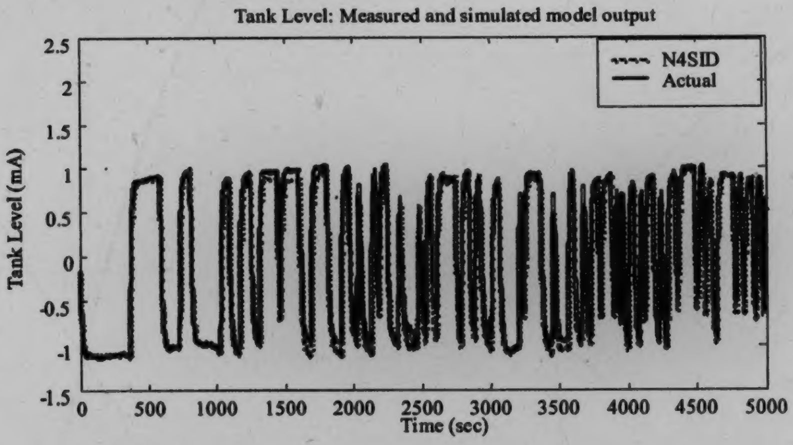


Figure 3.14: Water level: Comparison of model predictions with measured output

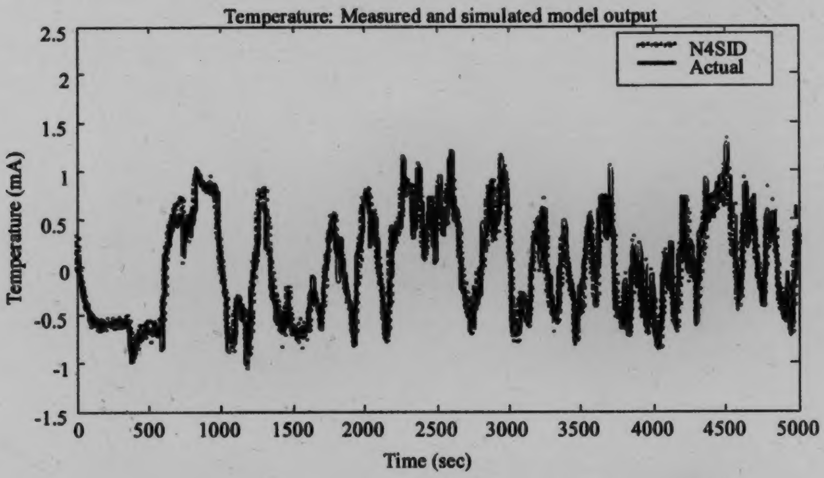


Figure 3.15: Water Temperature: Comparison of model predictions with measured output

3.4.1 Expermimental evaluation on a pilot-scale Continuous-flow Stirred Tank Heater (CSTH) process

Consider the schematic of an experimental setup of a Continuous-flow Stirred Tank Heater (Fig. 3.12). In this system, the tank level is controlled using the cold water valve position and the temperature of water in the tank is controlled using the steam valve position.

When there is a bias in the temperature sensor under closed-loop conditions, it is corrected by the controller and is not discernable from the trend plots of the sensor reading (Fig. 3.13).

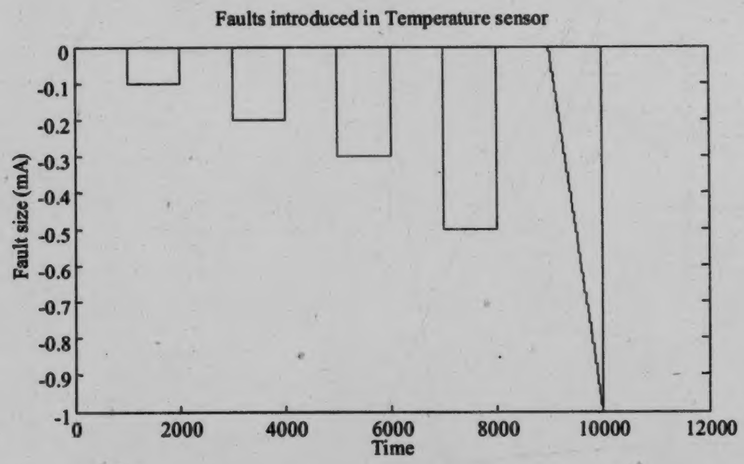


Figure 3.16: Temperature Sensor Faults introduced under closed-loop conditions

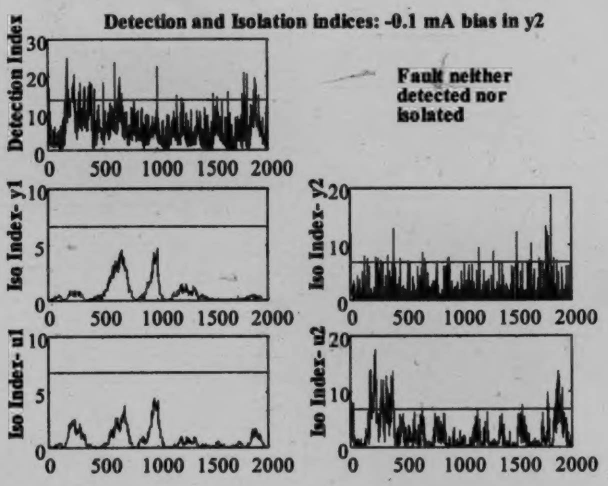


Figure 3.17: SRV-based FDI indices for -0.1mA sensor bias under closed-loop conditions

While this fault is not detected by looking at the trend plot, it can be detected and successfully isolated using model-based methods. One of the problems of using correlation-model-based methods where the overall model of the process is not

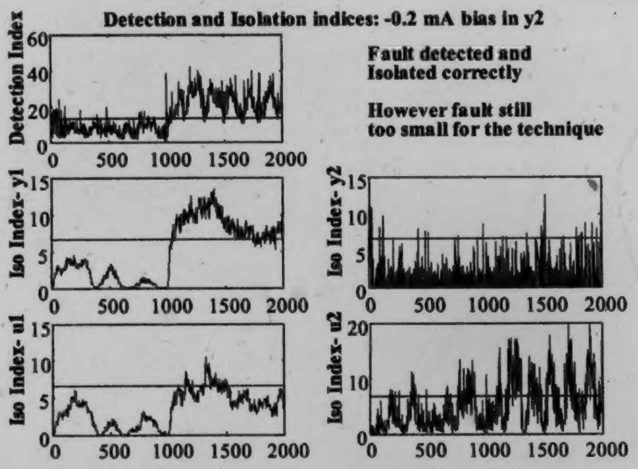


Figure 3.18: SRV-based FDI indices for -0.2mA sensor bias under closed-loop conditions

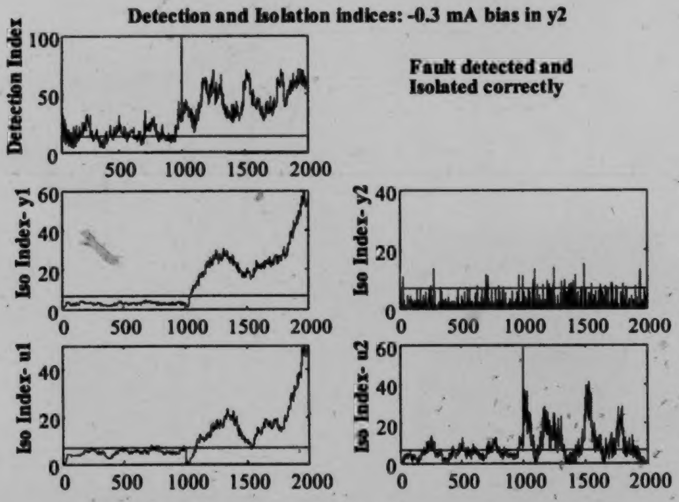


Figure 3.19: SRV-based FDI indices for -0.3mA sensor bias under closed-loop conditions

decomposed explicitly into the process and the controller is that it is not easy to separate the effect of the fault on particular sensors, *i.e.*, it quickly spreads to the other components in the system. Hence the location of the fault cannot be accurately isolated. However, when the open-loop model is available explicitly, this does not pose a problem. In this case-study, we have applied model-based methods to detect and isolate sensor faults in a pilot-scale CSTH system. The method is based on a model identified using *N4SID* (Numerical state-space

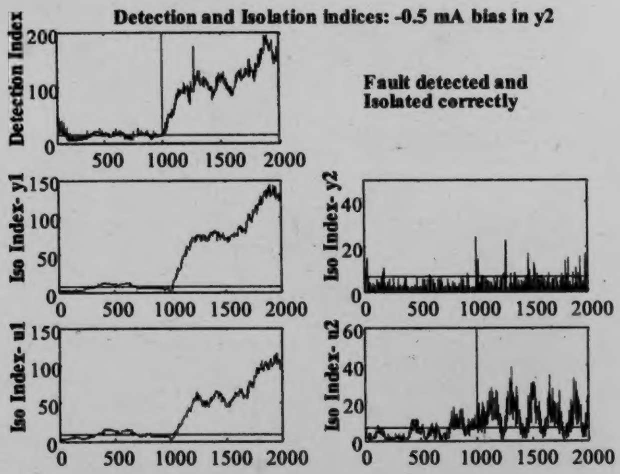


Figure 3.20: SRV-based FDI indices for -0.5mA sensor bias under closed-loop conditions

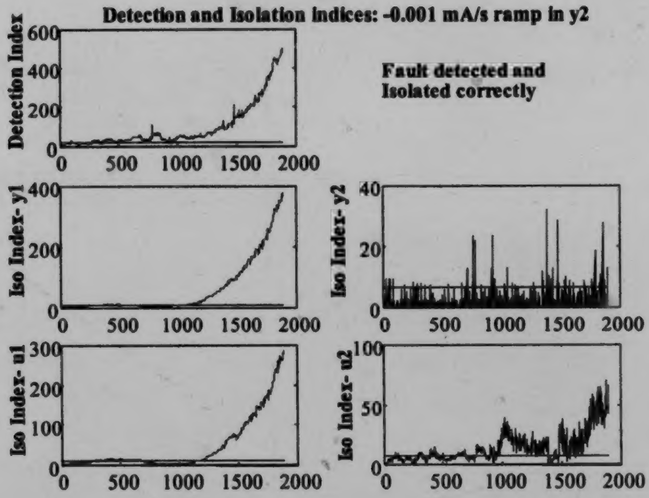


Figure 3.21: SRV-based FDI indices for ramp-type fault under closed-loop conditions

subspace identification), a subspace identification technique under open-loop conditions. Comparison plots of the predictions from the model and the actual process output under open-loop conditions, are given in Fig. 3.14 and 3.15.

Sensor faults were introduced into the temperature sensor under closed-loop conditions according to the Fig. 3.16.

Diagnosis of Actuator Biases using PCA under open-loop conditions		
Bias level	Results of Diagnosis	Comments
σ	Detection: T^2 detects change in operating condition	Fault affects only y1, y2 and y3.
2σ	Larger change in operating condition detected.	Clear detection.
3σ	T^2 way out of confidence bounds.	SPE does not cross bounds
6σ	SPE starts reacting.	Contribution plots are unreliable.

Table 3.2: *Open loop Actuator Bias diagnosis using PCA*

Diagnosis of Sensor Biases using PCA under open-loop conditions		
Bias level	Results of Diagnosis	Comments
σ	Not detected in SPE or T^2.	Fault too small Affects only y1
2σ	Statistically detected – 3.6% of points violate SPE bounds.	Detection not clear
3σ	Statistically detected – 7% of points violate SPE bounds.	Fault not easily discernable from trend plots
6σ	Clear detection from SPE plot. Isolation is unreliable.	Contribution plots are unreliable.

Table 3.3: *Open loop Sensor Bias diagnosis using PCA*

The results of the SRV-based detection and isolation are presented in Figs. 3.17 to 3.20. It is seen that a $-0.1mA$ bias is too small for the SRV method to detect and isolate clearly (Fig. 3.17). However, the method is able to detect and isolate biases $-0.2mA$ and larger (Figs. 3.18, 3.19 and 3.20). It is also able to detect and isolate a ramp-type fault (Fig. 3.21). This experiment confirms that it is possible to detect and isolate I/O faults using model-based techniques even under closed-loop conditions.

3.5 I/O fault diagnosis in a Quadruple tank process

We now present the results of sensor and actuator fault diagnosis using PCA and the SRV-based method under both open-loop and closed-loop conditions. The process description, fault introduction and sizes of various faults were discussed in sec. 3.2.

3.5.1 Fault diagnosis using PCA

PCA-based model identification and fault diagnosis were performed using the methodology presented in Secs. 2.2 and 2.4.

Diagnosis of Actuator Biases using PCA under closed-loop conditions		
Bias level	Results of Diagnosis	Comments
σ	Not detected in SPE or T^2.	Fault too small Affects only u1
2σ	Not detected in SPE or T^2.	Fault too small
3σ	Not detected in SPE or T^2.	Fault not discernable from trend plots
6σ	SPE & T^2 statistic detect fault statistically.	Fault discernable from trend plots

Table 3.4: *Closed loop Actuator Bias diagnosis using PCA*

Diagnosis of Sensor Biases using PCA under closed-loop conditions		
Bias level	Results of Diagnosis	Comments
σ	Not detected in SPE or T^2.	Fault too small
2σ	SPE & T^2 statistic detect fault statistically.	Fault not discernable from trend plots.
3σ	SPE & T^2 statistic detect fault statistically.	Fault discernable from trend plots.
6σ	SPE & T^2 statistic detect fault clearly	Fault discernable from trend plots.

Table 3.5: *Closed loop Sensor Bias diagnosis using PCA*

The results of PCA-based diagnosis are tabulated in Tables 3.2 to 3.5.

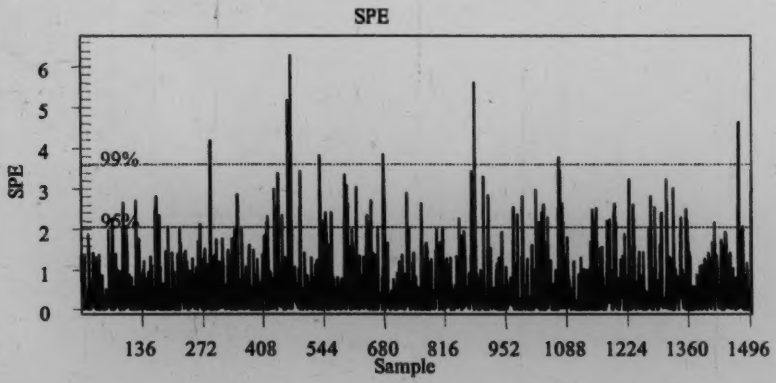


Figure 3.22: SPE plot for open-loop 1σ actuator bias in u_1

Fig. 3.22 shows that the SPE plot fails to detect a 1σ actuator bias under open-loop conditions, while this can be detected as a change in the operating condition as shown in Fig. 3.23. A larger actuator bias can be detected as a change in the operating condition

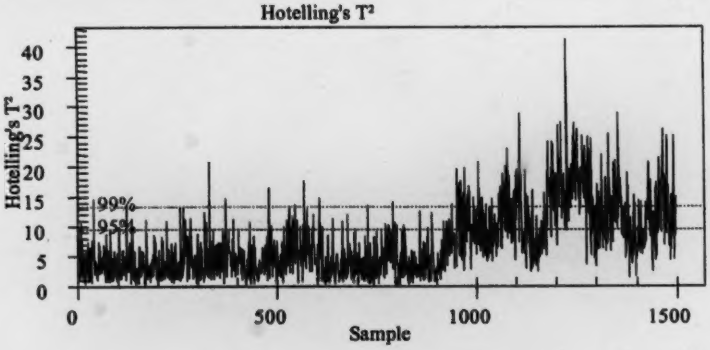


Figure 3.23: T^2 plot for open-loop 1σ actuator bias in u_1

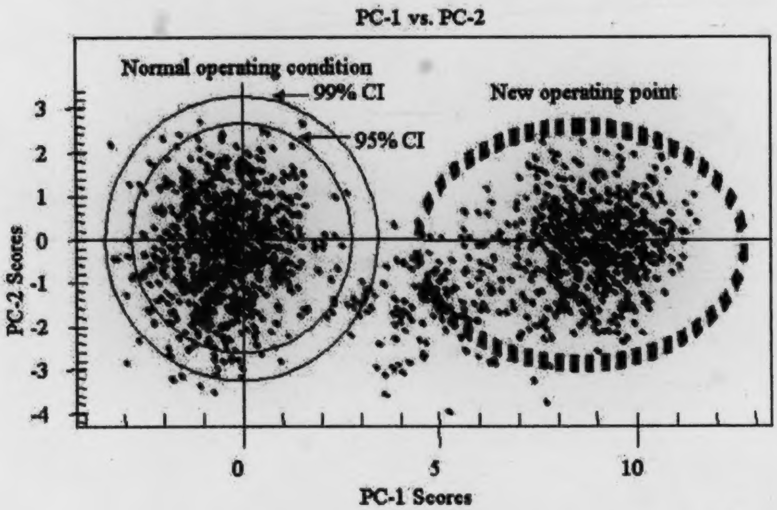


Figure 3.24: 2-D score plot for 3σ open loop actuator bias in u_1

from the T^2 plot and is also clear by looking at the two dimensional score plot as shown in Fig. 3.24.

Fig. 3.25 shows that the SPE plot clearly detects a 6σ sensor fault. However, the isolation of this fault is unreliable as shown in Fig. 3.26.

The SPE and T^2 plots are unable to detect small actuator faults under closed loop conditions. However, larger actuator faults are detected clearly in both these plots. Figs. 3.27 and 3.28 clearly show the detection of a 6σ actuator fault under closed-loop conditions. However, the isolation results using contribution plots is unreliable.

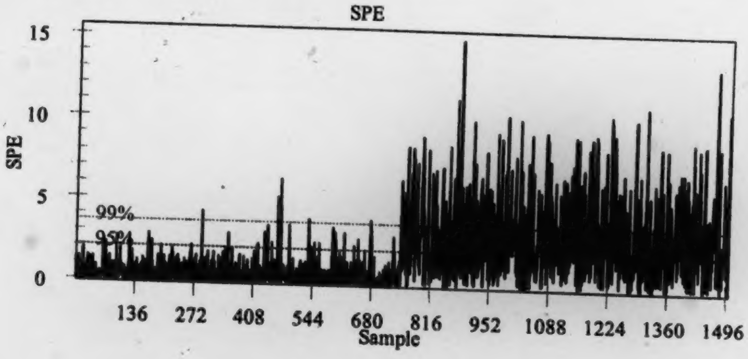


Figure 3.25: SPE plot for 6σ open loop sensor bias in y_1

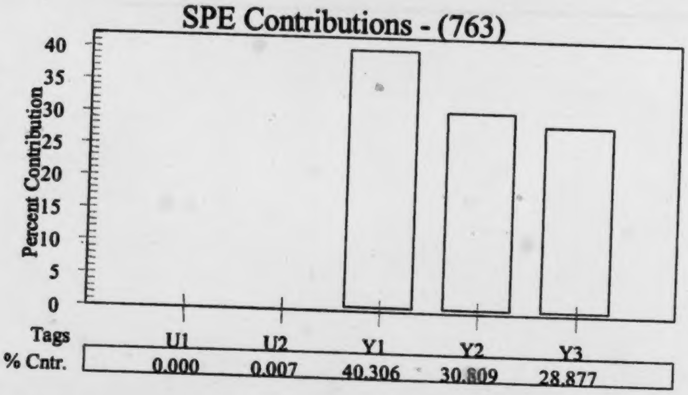


Figure 3.26: Contribution plot for 6σ open loop sensor bias in y_1

The SPE and T^2 plots clearly detect sensor faults even under closed loop conditions. Figs. 3.29 and 3.30 clearly show the detection of a 3σ sensor fault under closed loop conditions.

It is clear that PCA has good fault detection capabilities. These capabilities can be further enhanced by low-pass filtering using EWMA filters or by multiscale resolution using wavelet-based filters. However, fault isolation capabilities of contribution plots are poor because they only indicate the variables that are most correlated with the fault and this cannot be used to provide an unequivocal decision regarding the location of the fault. On the other hand, the structured residual approaches which use residual directions based on the identified model have powerful isolation properties. We illustrate this in the following section in which we present the results of SRV-based fault detection and diagnosis.

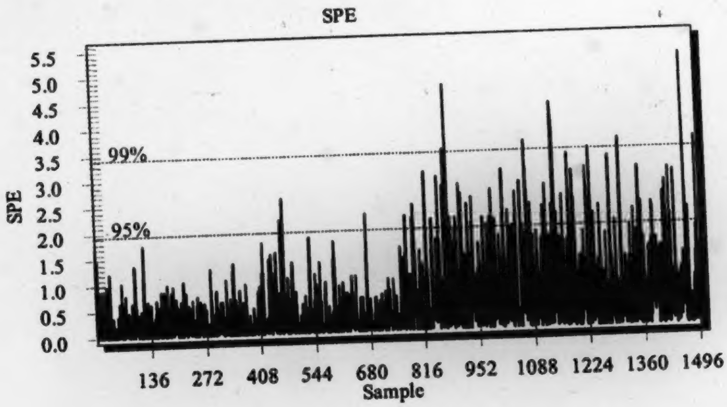


Figure 3.27: SPE plot for 6σ closed loop actuator bias in u_1

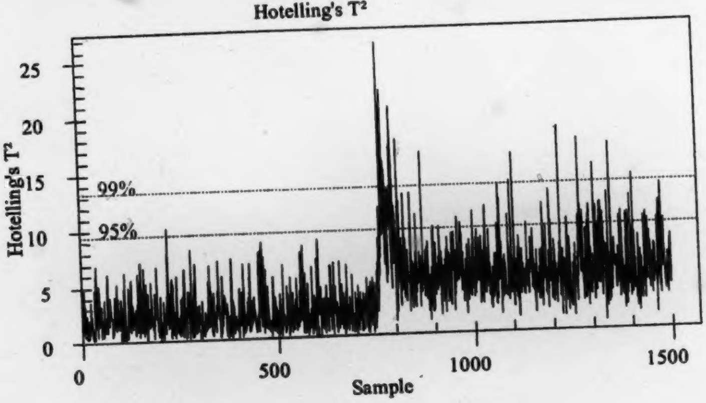


Figure 3.28: T^2 plot for 6σ closed loop actuator bias in u_1

3.5.2 Fault diagnosis using the SRV approach

SRV-based fault detection and diagnosis was performed using the methodology presented in Sec. 3.3. The fault detection index was constructed based on the Primary Residual Vector. In addition, 5 fault isolation indices insensitive to faults in the sensors and actuators, $[y_1, y_2, y_3, u_1, u_2]$ were constructed. The following figures show plots of the fault detection and isolation indices. A fault is present in a sensor or actuator if its isolation index is within the confidence bounds and all other isolation indices are outside their confidence bounds.

Fig. 3.31 shows that the SRV-based approach is able to successfully detect actuator faults under open-loop conditions. Fig. 3.32 shows the successful isolation of a 1σ actuator fault

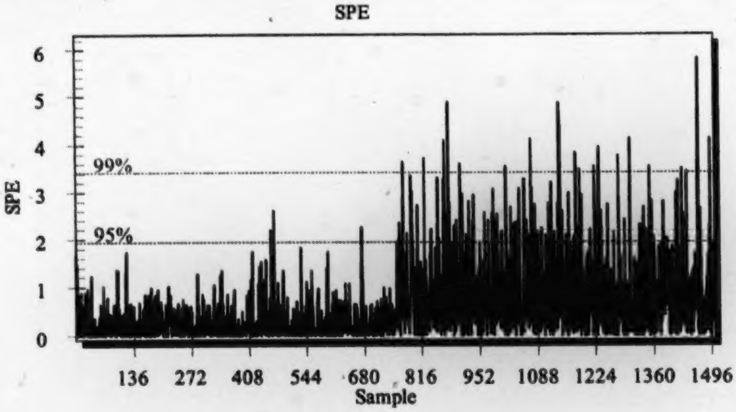


Figure 3.29: SPE plot for 3σ closed loop sensor bias in y_1

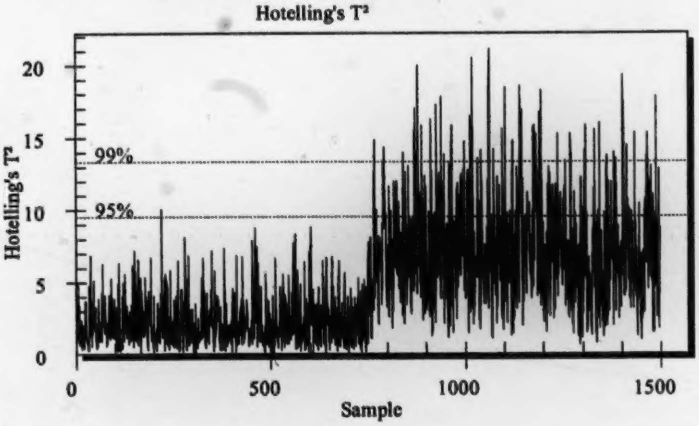


Figure 3.30: T^2 plot for 3σ closed loop sensor bias in y_1

in u_2.

Fig. 3.33 shows that the SRV-based approach is able to successfully detect sensor faults under open-loop conditions. Fig. 3.34 shows the successful isolation of a 3σ sensor fault in y_2. Sensor faults smaller than this are not successfully isolated by this approach.

Fig. 3.35 shows that the SRV-based approach is able to successfully detect actuator faults under closed-loop conditions. Fig. 3.36 shows the successful isolation of a 1σ actuator fault in u_1.

Fig. 3.37 shows that the SRV-based approach is able to successfully detect sensor faults under closed-loop conditions. Fig. 3.38 shows the successful isolation of a 3σ sensor fault

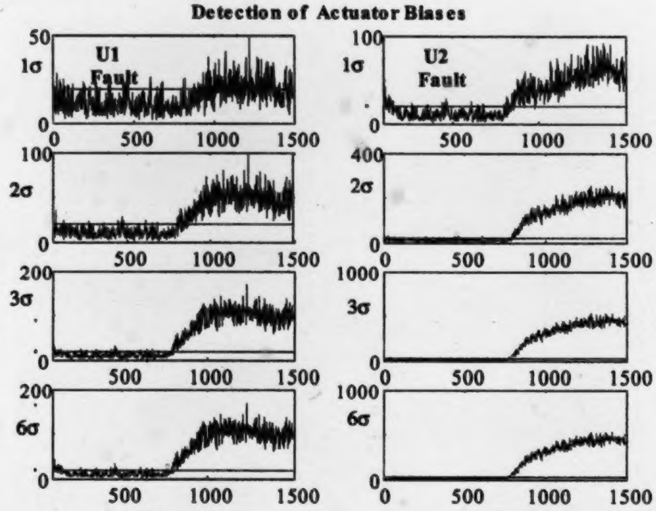


Figure 3.31: Detection of open-loop actuator faults using the SRV approach

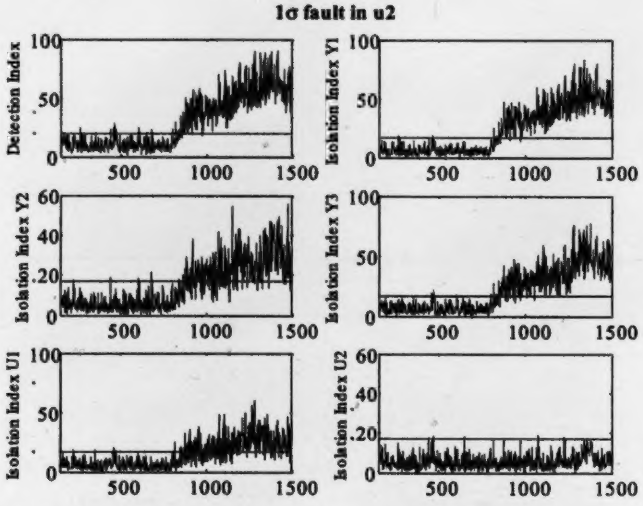


Figure 3.32: Isolation of 1σ actuator fault in u_2

in y_1. Sensor faults smaller than this are not successfully isolated by this approach.

These results show that the SRV-based approach can be successfully used for detection and isolation of sensor and actuator biases under open and closed-loop conditions.

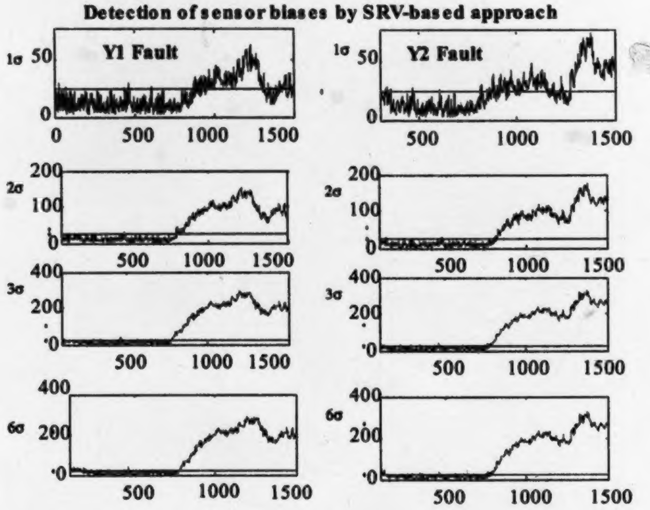


Figure 3.33: Detection of open-loop sensor faults using the SRV approach

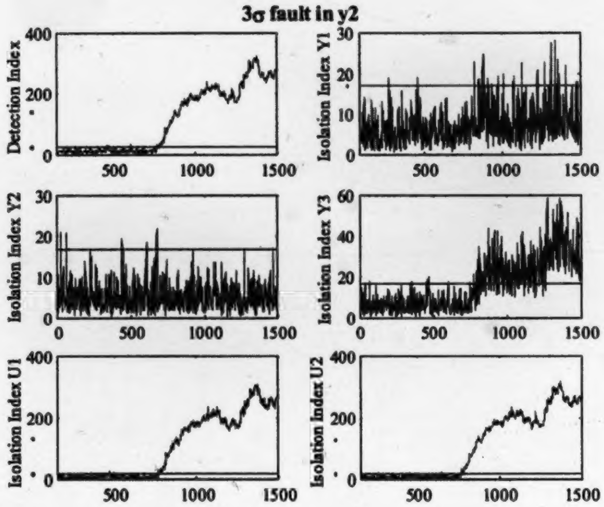


Figure 3.34: Isolation of 3σ sensor fault in y_2

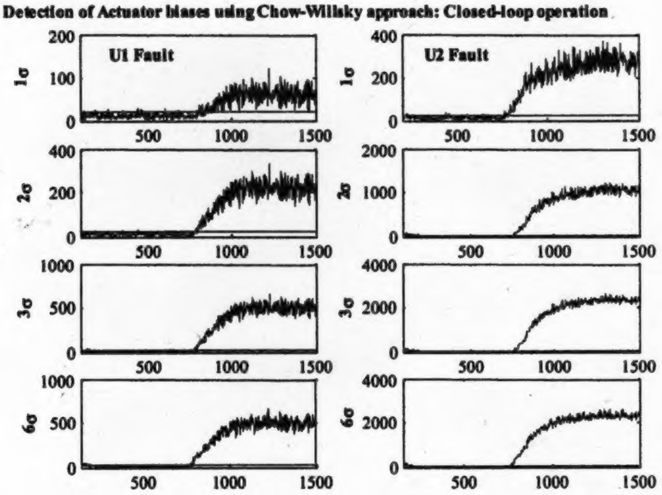


Figure 3.35: Detection of closed-loop actuator faults using the SRV approach

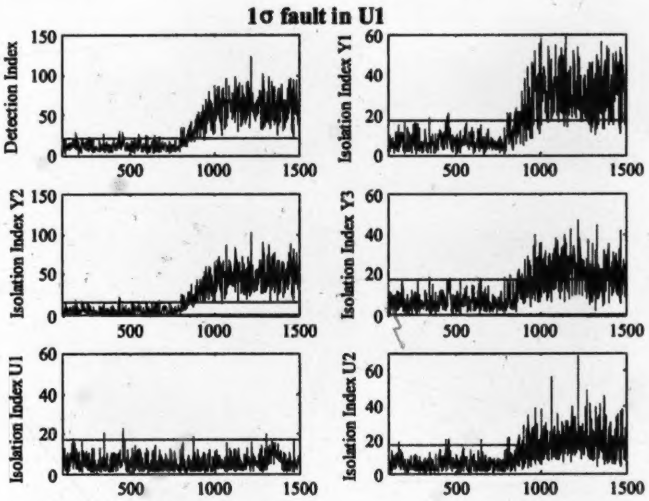


Figure 3.36: Isolation of 1σ actuator fault in u_1

3.6 Conclusions

Consequences of the occurrence of actuator and sensor faults in systems operating under open-loop and closed-loop conditions have been demonstrated using simulations on a

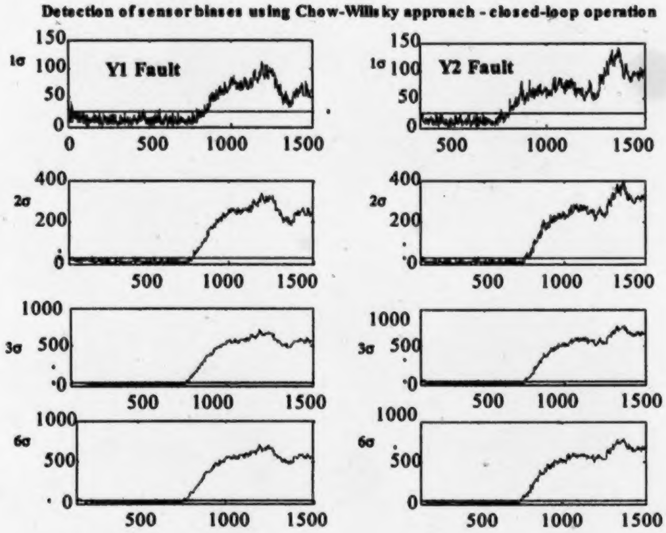


Figure 3.37: Detection of closed-loop sensor faults using the SRV approach

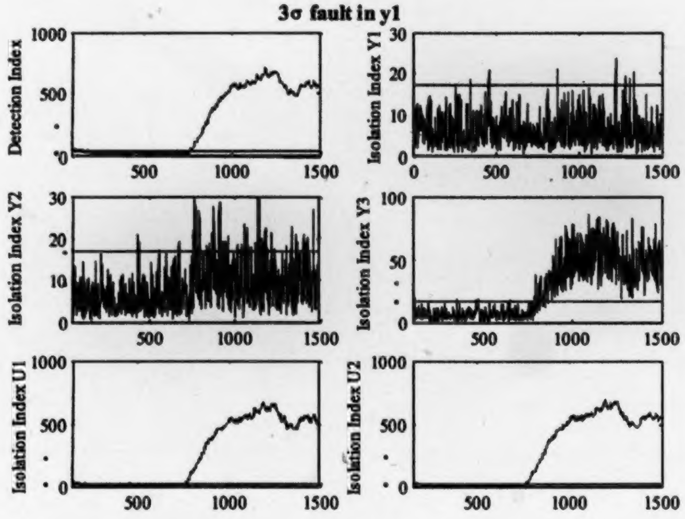


Figure 3.38: Isolation of 3σ sensor fault in y_1

quadruple tank process. The applicability of the SRV-based approach for detection and isolation of sensor and actuator faults under closed-loop conditions has been demonstrated using laboratory experiments and simulation case-studies. In addition, the superiority of the SRV-based isolation scheme over contribution plots traditionally associated with PCA, has also been demonstrated.

4

Identification of processes with irregular output sampling

4.1 Overview

In many chemical processes, variables which indicate product quality are infrequently and irregularly sampled. Often, the inter-sample behavior of these quality variables can be inferred from manipulated variables and other process variables which are measured frequently. Such an inferential mechanism can be developed using a mathematical model which relates the manipulated variables and fast-sampled process variables with the slowly sampled quality variables. When the quality variables are irregularly sampled, Maximum Likelihood Estimation (MLE) of the model parameters can be performed using the Expectation Maximization (EM) approach[1]. A state-space model identification procedure based on the EM algorithm yields a Kalman filter-based prediction-correction mechanism which can be used for optimal prediction of the quality variables. In this chapter we describe such a state-space model identification and estimation method. Further, we present the results of its application on simulation, laboratory-scale and industrial case studies.

---

[1] Sections of this chapter have been presented or published as:

1. Raghavan, H., A.K. Tangirala, R.B. Gopaluni and S.L. Shah. "Identification of chemical processes with irregular output sampling", submitted to Automatica, Jan 2004.
2. Gopaluni, R.B., H. Raghavan and S.L. Shah, "System Identification from multi-rate data", In Proc. AdCHEM-2003, Hongkong, Jan 2004.
3. Gopaluni, R.B., H. Raghavan and S.L. Shah. "An Iterative approach for the Identification of Multi-Rate Chemical Processes", In Proc. International Symposium on Process Systems Engineering and Control (ISPSEC'03), Mumbai, India, Jan 2003.

4.2 Introduction

In many chemical engineering applications, outputs may not be available as frequently as would be desired for satisfactory closed-loop control. For example, key product quality variables are available after several hours of lab analysis. The relationship used to predict quality variables from other process variables is often called an inferential sensor. It is based on state estimation and has many applications including inferential control and process monitoring (Li *et al.* 2002). There have been many reported inferential sensor applications based on static models identified using data-driven techniques such as principal components analysis and partial least squares (Kresta *et al.* 1994, Raghavan *et al.* 2003). Alternatively, it might be possible to identify dynamic models using system identification techniques, and design state estimators using these dynamic models.

Traditional system identification techniques for sampled-data systems with uniformly spaced sampling intervals include, Maximum Likelihood Estimation (Åström 1980), the closely related Prediction Error Methods (PEM) (Ljung 1999), Instrumental Variable Techniques (Söderström and P. Stoica 1983) and Subspace Identification (Overschee and DeMoor 1996). MLE and PEM have been popular because of their well established theoretical properties including asymptotic statistical optimality in terms of the achievement of the Cramér-Rao lower bound and well researched practical issues such as variance and bias distributions. However, MLE suffers from a number of practical problems. In general, MLE may involve solving a non-convex optimization problem. This is usually tackled using a gradient-based iterative search strategy. Based on observed, input-output data, the state-space model of a system can only be identified correct to a similarity transformation (Kailath 1980). The shape of the likelihood-based objective function depends on the chosen parameterization of the state-space matrices. When canonical state-space parameterizations are chosen the iterative techniques based on gradient-based search can suffer from serious numerical issues (Deistler 2000). In practice the popularly used PEM algorithm (*Matlab© SysID toolbox*) chooses a *free* parameterization for the state-space matrices, i.e., the elements in the matrices are freely adjustable by the estimation routines, though it can be argued that the state-space matrices should not be *filled* with parameters since the corresponding input-output description contains fewer number of parameters ($3n$ parameters in the input-output model compared with n^2+2n parameters for the state-space model for a single-input, single-output system). On the other hand, Subspace Identification techniques avoid the problem of canonical state-space parameterizations and identify state-space models in an arbitrary state-space basis which is a function of weighting matrices (Overschee and DeMoor 1996). An important advantage of these techniques is that, they are non-iterative and they are implemented using efficient algorithms such as SVD and QR-decomposition. However, the statistical properties of these techniques *vis-à-vis* MLE, have not yet been established.

In most chemical processes three classes of measurements can be distinguished (Amirthalingam *et al.* 2000, Ergon 1998); input variables which can be manipulated at a "fast-rate" (*e.g.*, control-valve positions, $\mathbf{u}$), output variables measured at the fast-rate (*e.g.*, flow rates, temperatures, $\mathbf{y_1}$) and output variables measured at a "slow rate" (*e.g.*, compositions, $\mathbf{y_2}$). Such processes with differing sample rates for the measured variables are known as *multirate* processes and model identification for such processes is of interest to the chemical industry. The fastest sample rate is called the *base* sample rate and unavailable data points in the slow measurements are called *missing* data.

The problem of identifying optimal models when some of the variables are irregularly sampled has been studied in statistical literature using the Expectation Maximization (EM) approach (Dempster *et al.* 1977, Shumway and Stoffer 1982, Titterington 1984, Ninness and Gibson 2002, Gibson and Ninness 2000). The EM approach has also been used for dynamic data rectification using its state estimation properties, in chemical engineering literature (Singhal and Seborg 2000).

An excellent overview of the use of various techniques for model identification subject to missing data, including traditional MLE and the EM-algorithm, is provided in Isaksson (1993). The traditional gradient-based MLE algorithm has been implemented by Isaksson (1993) with the observed data likelihood function calculated using a modified form of the Kalman filter (Jones 1980, Ansley and Kohn 1983) to account for missing observations. This Kalman filter uses non-constant state-space matrices, the time-varying nature of which is dictated by the missing observations. In addition, Isaksson (1993) uses a quasi-Newton based method with secant update for the Hessian along with a forward-difference approximation for the derivatives, with the comment that the numerical optimization method for this problem is quite costly. The presence of missing-data can aggravate the numerical problems faced by gradient-based iterative search strategies traditionally used to solve the MLE problem. While the gradient-based techniques demonstrate faster convergence than the EM algorithm in general, Isaksson (1993) shows through simulation that the EM can be much faster than the gradient-based techniques when there is a significant amount of missing data. While, the work in Isaksson (1993) is restricted to the ARX model structure, we present the model identification problem in the presence of missing data for the more general state-space class of models in this chapter.

A commonly used approach for the identification of multirate processes is to interpolate between available sampled data. Techniques such as linear or quadratic interpolation are used (Amirthalingam *et al.* 2000). However, these arbitrary interpolations are usually univariate and do not take the multivariate dynamic nature of the process into account. As a result they can lead to sub-optimal models and unreliable predictions.

Another possible approach to deal with system identification from multirate data is to use the so-called "direct" continuous-time (CT) model identification. Direct methods attempt to identify CT models from discrete-time (DT) signals directly, in contrast to indirect methods which identify DT models first and then convert them to equivalent CT models. However, there are a number of issues in CT identification from discrete-time (DT) signals. In comparison with DT model identification, CT model identification involves differentiation of the input and output signals (Wang and Gawthrop 2001). This differentiation can be implemented using finite differences (Söderström *et al.* 1997) with the disadvantage that these operations generally enhance the strength of the noise in the signals. Alternatively, these differentiation operations can be avoided using techniques such as linear filters, integral methods and modulating function methods (Young 1981, Unbehauen and Rao 1990, Sinha and Rao 1991). While there has been a considerable amount of research for identifying SISO CT models using these approaches, work on MIMO CT model identification is in its infancy. For example, choosing the order of the finite horizon multiple integral filter (Sagara and Zhao 1990, Kowalczuk and Kozłowski 2000) for MIMO identification and the identification of the CT noise model are non-trivial tasks (Wang and Gawthrop 2001). Although there has been some progress in establishing the statistical properties of the identified CT model parameters for AR and ARMA processes (Söderström 1999, Larsson and Larsson 2002, Larsson and Larsson 2004) the more general state-space model has not yet been considered.

There have also been attempts at solving the multi-rate identification problem using lifting techniques (Li *et al.* 2001*a*). The lifting operator (Kranc 1957, Freidland 1961, Khargonekar *et al.* 1985, Chen and Francis 1995) is used to rearrange the data and convert the multi-rate identification problem into a slow-single rate identification problem with increased dimensionality. However, there are a few issues here. Firstly, in many chemical processes there is a large difference in the sampling rates and this quickly leads to unmanageable dimensionality in the *lifted* model. Secondly, converting the lifted model to the base sampling rate is non-trivial (Li *et al.* 2001*a*, Wang *et al.* 2004). Furthermore, this method cannot handle irregularly sampled data.

The focus of this chapter is on identifying dynamic models of processes with irregular output sampling using the EM algorithm. The problem of identifying state-space models using the EM approach has been addressed in Ninness and Gibson (2002), but the authors have not considered the missing-data case. Also, while Shumway and Stoffer (1982) and Shumway and Stoffer (2000) consider the application of the EM algorithm for state-space model identification in the presence of missing data, they outline the procedure only for the case with no manipulated inputs, i.e., the time-series model identification problem. A similar procedure is outlined in Tanaka and Katayama (1990) for state-space model identification for processes without manipulated inputs, where the focus is on identification in the presence of outliers which can be treated as missing-data. In addition, in these articles, there are no guidelines on choosing the parameters for the initial model though the authors state that this is an important step in the MLE procedure. In Isaksson (1993), the author restricts his attention to the ARX model structure.

In this chapter, we perform state-space model identification from datasets which have irregularly sampled outputs. We use a realization-based subspace identification technique to obtain an initial model (Kung 1978, Viberg 1995). The advantage of this technique is that it involves a numerically robust SVD implementation without iterative identification. This procedure is also used to fix the order of the state-space model using a plot of the singular values, and the resulting state-space matrices have a *free* parameterization. This initial model is used in the EM algorithm, modified from Shumway and Stoffer (2000) to include manipulated inputs, to obtain estimates of the state-space matrices, conditioned on all available input-output data. This procedure can handle both irregularly sampled and regularly sampled multirate output data and is appropriate for most chemical processes. Expressions for the state-space matrices obtained in the M-step, in the presence of irregular observations, are provided. The Kalman filter obtained as a part of the identification process is used for predicting the future state and output trajectory. The model parameters and predictions obtained are maximum likelihood estimates when the random vectors associated with the system have Gaussian densities. In the non-Gaussian case, the state estimates obtained using the Kalman filter are minimum mean-squared error estimates within the class of linear estimators (Shumway and Stoffer 2000).

The rest of this chapter is as follows. In Sec. 4.3, we present the model structure with a brief description of the identification and estimation problems. Following this, we present the EM-based state-space model identification procedure in Sec. 4.4 and the modifications required to handle irregular output measurements in Sec. 4.6. Sec. 4.5 contains the description of a subspace identification method which is used to obtain the initial model for the EM-procedure. In Sec. 4.7, we present the results obtained on applying the proposed identification procedure to simulation, laboratory-scale and industrial case studies.

4.3 Problem description

4.3.1 Model Structure and Assumptions

Let us assume that the process is represented by the discrete-time state-space model shown in Eq. 4.1:

$$\begin{aligned} \mathbf{x}_t &= \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1} + \mathbf{w}_t \\ \mathbf{y}_t &= \mathbf{C}\mathbf{x}_t + \mathbf{v}_t \end{aligned} \tag{4.1}$$

where, $\mathbf{x}_t \in \Re^n$ is the state vector, $\mathbf{u}_t \in \Re^m$ is the manipulated input, $\mathbf{y}_t \in \Re^p$ is the output, and $\mathbf{w}_t \in \Re^n$ and $\mathbf{v}_t \in \Re^p$ are the state and measurement noise vectors respectively. We assume that the initial state and the noise vectors are *i.i.d* random variables with a normal distribution. $\mathbf{x}_0 \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0), \mathbf{w}_t \sim N(\mathbf{0}, \mathbf{R}_w), \mathbf{v}_t \sim N(\mathbf{0}, \mathbf{R}_v)$. For simplicity we assume that the noise vectors are mutually uncorrelated and are also uncorrelated with the initial state. Further, the outputs can be classified as fast-sampled outputs (secondary outputs, $\mathbf{y}_1 \in \Re^{p_1}$) and slow-sampled outputs (primary outputs, $\mathbf{y}_2 \in \Re^{p_2}$). Hence,

$$\mathbf{y}_t = \begin{bmatrix} \mathbf{y}_{1,t} \\ \mathbf{y}_{2,t} \end{bmatrix}, \; \mathbf{C} = \begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \end{bmatrix}, \; \mathbf{v}_t = \begin{bmatrix} \mathbf{v}_{1,t} \\ \mathbf{v}_{2,t} \end{bmatrix} \tag{4.2}$$

We assume that the fast-sample interval (T_{s_1}) is fixed, while the slow-sample interval (T_{s_2}) is variable corresponding to the irregular sampling case. In addition, we assume that $T_{s_2} = m_l \times T_{s_1}, \; m_l \in Z^+$. We also assume that the primary output measurement noise is uncorrelated with the secondary output measurement noise.

$$\mathbf{R}_v = \begin{bmatrix} \mathbf{R}_{v1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{v2} \end{bmatrix} \tag{4.3}$$

We define, $\mathbf{Y}_s \equiv \{\mathbf{y}_1, \ldots, \mathbf{y}_s\}$, $\mathbf{U}_s \equiv \{\mathbf{u}_1, \ldots, \mathbf{u}_s\}$ and $\mathbf{Z}_s \equiv \{\mathbf{Y}_s, \mathbf{U}_s\}$. Further, we use the following definitions for the conditional expectations of the states and the corresponding error covariances:

$$\begin{aligned} \mathbf{x}_t^s &= E\left(\mathbf{x}_t|\mathbf{Z}_s\right) \\ \mathbf{P}_{t_1,t_2}^s &= E\left((\mathbf{x}_{t_1} - \mathbf{x}_{t_1}^s)(\mathbf{x}_{t_2} - \mathbf{x}_{t_2}^s)^T\right) \\ &= E\left((\mathbf{x}_{t_1} - \mathbf{x}_{t_1}^s)(\mathbf{x}_{t_2} - \mathbf{x}_{t_2}^s)^T|\mathbf{Z}_s\right) \end{aligned} \tag{4.4}$$

where $E(\,)$ refers to the expectation operator. For convenience, when $t_1 = t_2 = t$, $\mathbf{P}_{t_1,t_2}^s$ is written as $\mathbf{P}_t^s$.

With these assumptions, the two main objectives of this work are model identification and optimal estimation of the primary outputs from observed data. We described these as follows:

4.3.2 Identification problem

In the identification problem we are interested in obtaining optimal estimates for the model parameters $\hat{\mathbf{\Theta}} \equiv (\boldsymbol{\mu}_0, \mathbf{\Sigma}_0, \mathbf{A}, \mathbf{B}, \mathbf{C}_1, \mathbf{C}_2, \mathbf{R}_w, \mathbf{R}_{v1}, \mathbf{R}_{v2})$ using all the available samples in the identification data set. The data comprises N equally spaced samples of $\mathbf{u}$ and $\mathbf{y}_1$ and N_o unequally spaced samples of $\mathbf{y}_2$, $N_o \leq N$. For optimal estimation we set-up this problem as an MLE problem and solve it using an iterative EM-based approach.

4.3.3 Estimation problem

In the estimation problem we are interested in obtaining optimal estimates of the primary output vector $\mathbf{y}_2$ at every sampling instant using all the information available up to and including that time instant. At this sampling instant, the primary output may or may not be measured. Consequently, the optimal estimation mechanism uses the manipulated inputs and secondary outputs to predict the primary output when it is not measured and the primary output value when it is measured.

4.4 EM-based state space model identification

MLE-based identification algorithms and closely related PEM algorithms are usually implemented using gradient-based numerical optimization techniques. The EM algorithm provides an alternative to these gradient-based MLE algorithms. It is an iterative estimation technique which was developed by Dempster *et al.* (1977) mainly for MLE from data sets with missing observations. However, the EM algorithm can also be used for state-space model identification (Shumway and Stoffer 1982). We briefly describe EM-based state-space identification after a short summary of the traditional MLE approach for state-space model identification. Let us momentarily assume in this section, that all the data records are complete and that there are no missing observations. We will relax this assumption later when we describe the modifications necessary to handle missing data in the EM algorithm.

4.4.1 Maximization of the likelihood function

The parameters of the state-space model shown in Eq. 4.1 can be estimated by maximizing the *likelihood* function of the observed data ($\mathbf{Z}_N \equiv \{\mathbf{Y}_N, \mathbf{U}_N\}$). The likelihood function can be written in terms of *innovations* or *one-step-ahead* prediction errors (Ljung 1999) which are defined as follows:

$$\epsilon_t \equiv \mathbf{y}_t - E\left(\mathbf{y}_t|\mathbf{Z}_{t-1}\right) = \mathbf{y}_t - \mathbf{C}\mathbf{x}_t^{t-1} \tag{4.5}$$

Under the distributional assumptions stated in the previous section for the initial state and the noise processes, $\epsilon_t \sim N(0, \mathbf{\Sigma}_t)$, $\mathbf{\Sigma}_t = \mathbf{C}\mathbf{P}_t^{t-1}\mathbf{C}^T + \mathbf{R}_v$. The innovations form of the likelihood function (Schweppe 1965) can be written as:

$$L(\mathbf{\Theta}|\mathbf{Z}_N) = \prod_{t=1}^{N} \frac{1}{(2\pi)^{\frac{p}{2}}|\mathbf{\Sigma}_t(\mathbf{\Theta})|^{\frac{1}{2}}} \exp\left(\frac{-\epsilon_t(\mathbf{\Theta})^T\mathbf{\Sigma}_t(\mathbf{\Theta})^{-1}\epsilon_t(\mathbf{\Theta})}{2}\right) \tag{4.6}$$

where, $|\mathbf{\Sigma}_t(\mathbf{\Theta})|$ refers to the determinant of the matrix $\mathbf{\Sigma}_t(\mathbf{\Theta})$. The elements of $\mathbf{\Theta}$ appear in this likelihood function as follows:

$$\begin{aligned}
\epsilon_1 &= \mathbf{y}_1 - \mathbf{C}\mathbf{A}\boldsymbol{\mu}_0 \\
\epsilon_t &= \mathbf{y}_t - \mathbf{C}\mathbf{A}\mathbf{x}_{t-1}^{t-1} - \mathbf{C}\mathbf{B}\mathbf{u}_{t-1}, \quad t = 2, \ldots, N \\
\mathbf{\Sigma}_1 &= \mathbf{C}\mathbf{A}\Sigma_0\mathbf{A}^T\mathbf{C}^T + \mathbf{C}\mathbf{R}_w\mathbf{C}^T + \mathbf{R}_v \\
\mathbf{\Sigma}_t &= \mathbf{C}\mathbf{A}\mathbf{P}_{t-1}^{t-1}\mathbf{A}^T\mathbf{C}^T + \mathbf{C}\mathbf{R}_w\mathbf{C}^T + \mathbf{R}_v, \quad t = 2, \ldots, N
\end{aligned} \tag{4.7}$$

The quantities $\mathbf{x}_t^t$ and $\mathbf{P}_t^t$ are calculated using the Kalman filter. In addition, we present the Kalman smoother and Lag-one covariance smoother recursions, which will be used later in the EM algorithm.

4.4.2 Kalman filter and smoother expressions

The Kalman filter is an optimal, recursive, linear estimator, which estimates the state $\mathbf{x}_t$ by weighting the measurements $\mathbf{Y}_s$ according to *a priori* information about their accuracies (Sorenson 1985). Depending on the relative values of s and t, the estimation is classified as prediction ($s < t$), filtering ($s = t$) or smoothing ($s > t$). We now present the recursive expressions for optimal state and covariance estimates corresponding to the situations,

$s = t-1$, $s = t$, $s = N$. The proofs for these can be derived on the lines of Shumway and Stoffer (2000).

Lemma 4.1 *Kalman Filter:* For the model defined in Eq. 4.1 and given initial conditions $\mathbf{x}_0^0 = \boldsymbol{\mu}_0$, $\mathbf{P}_0^0 = \boldsymbol{\Sigma}_0$,

$$\begin{aligned} &\mathbf{x}_t^{t-1} = \mathbf{A}\mathbf{x}_{t-1}^{t-1} + \mathbf{B}\mathbf{u}_{t-1}, \quad \mathbf{P}_t^{t-1} = \mathbf{A}\mathbf{P}_{t-1}^{t-1}\mathbf{A}^T + \mathbf{R}_w \\ &\mathbf{K}_t = \mathbf{P}_t^{t-1}\mathbf{C}^T[\mathbf{C}\mathbf{P}_t^{t-1}\mathbf{C}^T + \mathbf{R}_v]^{-1} \\ &\mathbf{x}_t^t = \mathbf{x}_t^{t-1} + \mathbf{K}_t(\mathbf{y}_t - \mathbf{C}\mathbf{x}_t^{t-1}), \quad \mathbf{P}_t^t = [\mathbf{I}_n - \mathbf{K}_t\mathbf{C}]\mathbf{P}_t^{t-1} \end{aligned} \tag{4.8}$$

Lemma 4.2 *Kalman Smoother:* For the model defined in Eq. 4.1 and with initial conditions $\mathbf{x}_N^N$ and $\mathbf{P}_N^N$ obtained from the Kalman filter, smoothed estimates for $t = N-1, N-2, \ldots, 1$ can be obtained by the recursions:

$$\begin{aligned} \mathbf{J}_{t-1} &= \mathbf{P}_{t-1}^{t-1}\mathbf{A}^T\left(\mathbf{P}_t^{t-1}\right)^{-1} \\ \mathbf{x}_{t-1}^N &= \mathbf{x}_{t-1}^{t-1} + \mathbf{J}_{t-1}\left(\mathbf{x}_t^N - \mathbf{x}_t^{t-1}\right) \\ \mathbf{P}_{t-1}^N &= \mathbf{P}_{t-1}^{t-1} + \mathbf{J}_{t-1}\left(\mathbf{P}_t^N - \mathbf{P}_t^{t-1}\right) \end{aligned} \tag{4.9}$$

Lemma 4.3 *Lag-one Covariance Smoother:* For the model defined in Eq. 4.1, with $\mathbf{J}_t, \mathbf{K}_t, t = 1, 2, \ldots, N$ and $\mathbf{P}_N^N$ obtained from the filter and smoother, we initialize:

$$\begin{aligned} &\mathbf{P}_{N,N-1}^N = \left(\mathbf{I}_n - \mathbf{K}_N\mathbf{C}\right)\mathbf{A}\mathbf{P}_{N-1}^{N-1} \\ &\text{For } t = N-1, N-2, \ldots, 1, \\ &\mathbf{P}_{t,t-1}^N = \mathbf{P}_t^t\mathbf{J}_{t-1}^T + \mathbf{J}_t\left(\mathbf{P}_{t+1,t}^N - \mathbf{A}\mathbf{P}_t^t\right)\mathbf{J}_{t-1}^T \end{aligned} \tag{4.10}$$

Maximum likelihood estimates of the model parameters are obtained by maximizing the likelihood function, *i.e.*, $\hat{\boldsymbol{\Theta}} = \arg\max_{\boldsymbol{\Theta}} L(\boldsymbol{\Theta}|\mathbf{Z}_N)$. Because of the monotonicity of the logarithmic function, the *log-likelihood* function, $\ell(\boldsymbol{\Theta}|\mathbf{Z}_N) = \log L(\boldsymbol{\Theta}|\mathbf{Z}_N)$, can be maximized.

Observation 4.1

$$\hat{\boldsymbol{\Theta}}_{MLE} = \arg\max_{\boldsymbol{\Theta}} L(\boldsymbol{\Theta}|\mathbf{Z}_N) = \arg\max_{\boldsymbol{\Theta}} \ell(\boldsymbol{\Theta}|\mathbf{Z}_N) \tag{4.11}$$

Proof: Let us assume that $\boldsymbol{\Theta}^*$ is a stationary point of the likelihood function, $L(\boldsymbol{\Theta}|\mathbf{Z}_N)$, and $\boldsymbol{\Theta}(k)$ is the k^{th} element of Vec$\{\boldsymbol{\Theta}\}$, obtained by applying the *vectorization* operation on $\boldsymbol{\Theta}$.

Then, $\left.\frac{\partial L(\boldsymbol{\Theta}|\mathbf{Z}_N)}{\partial \boldsymbol{\Theta}(k)}\right|_{\boldsymbol{\Theta}=\boldsymbol{\Theta}^*} = 0.$

Applying the chain rule, $\frac{\partial \ell(\boldsymbol{\Theta}|\mathbf{Z}_N)}{\partial \boldsymbol{\Theta}(k)} = \frac{\partial \log L(\boldsymbol{\Theta}|\mathbf{Z}_N)}{\partial L(\boldsymbol{\Theta}|\mathbf{Z}_N)} \frac{\partial L(\boldsymbol{\Theta}|\mathbf{Z}_N)}{\partial \boldsymbol{\Theta}(k)}$.

Since, $\frac{\partial \log L(\boldsymbol{\Theta}|\mathbf{Z}_N)}{\partial L(\boldsymbol{\Theta}|\mathbf{Z}_N)} = 0 \Rightarrow L \to \infty$, the stationary points of $\ell(\boldsymbol{\Theta}|\mathbf{Z}_N)$ are identical to the stationary points of $L(\boldsymbol{\Theta}|\mathbf{Z}_N)$. Hence Eq. 4.11 holds. ■

Alternatively, the negative log-likelihood function can be minimized:

$$J(\boldsymbol{\Theta}) = -2\ell(\boldsymbol{\Theta}|\mathbf{Z}_N) = pN\log(2\pi) + \sum_{t=1}^{N} \log|\Sigma_t(\boldsymbol{\Theta})| + \sum_{t=1}^{N} \left(\epsilon_t(\boldsymbol{\Theta})^T \Sigma_t(\boldsymbol{\Theta})^{-1} \epsilon_t(\boldsymbol{\Theta})\right) \tag{4.12}$$

In practice, due to the non-convexity of the optimization problem, it is customary to use numerical techniques for the solution (Gupta and Mehra 1974). For example, gradient-based iterative techniques use the iteration (Ljung 1999):

$$\boldsymbol{\Theta}^{(i+1)} = \boldsymbol{\Theta}^{(i)} - \rho^{(i)} \mathbf{R}^{(i)} \mathbf{g}^{(i)} \tag{4.13}$$

where, $\boldsymbol{\Theta}^{(i)}$ refers to the parameters at the i^{th} iteration, $\mathbf{g}^{(i)}$ is the vector of first partial derivatives of $J(\boldsymbol{\Theta})$ evaluated at $\boldsymbol{\Theta}^{(i)}$, $\mathbf{R}^{(i)}$ is related to the inverse of the matrix of second partial derivatives of $J(\boldsymbol{\Theta})$ and $\rho^{(i)}$ is a one-dimensional search parameter. While, the Newton-Raphson method involves explicit computation of the inverse of the matrix of second partial derivatives, the Gauss-Newton approximates it as:

$$\mathbf{R}^{(i)} = \left[\left(\frac{\partial J(\boldsymbol{\Theta})}{\partial \mathrm{Vec}\{\boldsymbol{\Theta}\}}\right)\left(\frac{\partial J(\boldsymbol{\Theta})}{\partial \mathrm{Vec}\{\boldsymbol{\Theta}\}}\right)^T\right]^{-1} \tag{4.14}$$

In addition, it is possible to use numerical finite difference approximations for these derivatives, instead of obtaining explicit analytical expressions.

Algorithm 4.1 The usual procedure for a gradient-based technique is as follows:

- Start with an initial guess for the model parameters, $\boldsymbol{\Theta}^{(0)}$.
- Generate filtered state estimates and covariances using the Kalman filter.
- Obtain updated estimates for $\boldsymbol{\Theta}$ using Eq. 4.13. This involves calculation of $\rho^{(i)}$, $\mathbf{R}^{(i)}$ and $\mathbf{g}^{(i)}$.
- Repeat this procedure until convergence, which can be monitored using the likelihood function (Eq. 4.6).

In order to account for missing data, the observed data likelihood function can be calculated using a modified Kalman filter (Jones 1980, Ansley and Kohn 1983). This Kalman filter uses non-constant state-space matrices, the time-varying nature of which is dictated by the missing observations.

4.4.3 Modified Kalman filter for missing data

To handle missing data, let us define, $\mathbf{y}_t^{obs} = \mathbf{D}_t\mathbf{y}_t$, where,

$$\mathbf{D}_t = \begin{cases} \mathbf{I}_p & \text{if } \mathbf{y}_{2,t} \text{ is measured} \\ \left[\begin{array}{c|c} \mathbf{I}_{p_1} & \mathbf{0} \end{array}\right] & \text{if } \mathbf{y}_{2,t} \text{ is missing} \end{cases} \tag{4.15}$$

where, $\mathbf{I}_p$ and $\mathbf{I}_{p_1}$ refer to the $p \times p$ and $p_1 \times p_1$ identity matrices respectively. Let $\mathbf{H}_t = \mathbf{D}_t\mathbf{C}$. The system, Eq. 4.1 can now be written as:

$$\begin{aligned} \mathbf{x}_t &= \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1} + \mathbf{w}_t \\ \mathbf{y}_t^{obs} &= \mathbf{H}_t\mathbf{x}_t + \mathbf{D}_t\mathbf{v}_t \end{aligned} \tag{4.16}$$

Lemma 4.4 The Kalman filter recursions for the model defined in Eq. 4.16 are given by,

$$\begin{aligned} &\mathbf{x}_0^0 = \boldsymbol{\mu}_0, \quad \mathbf{P}_0^0 = \boldsymbol{\Sigma}_0 \\ &\mathbf{x}_t^{t-1} = \mathbf{A}\mathbf{x}_{t-1}^{t-1} + \mathbf{B}\mathbf{u}_{t-1}, \quad \mathbf{P}_t^{t-1} = \mathbf{A}\mathbf{P}_{t-1}^{t-1}\mathbf{A}^T + \mathbf{R}_w \\ &\mathbf{K}_t = \mathbf{P}_t^{t-1}\mathbf{H}_t^T[\mathbf{H}_t\mathbf{P}_t^{t-1}\mathbf{H}_t^T + \mathbf{D}_t\mathbf{R}_v\mathbf{D}_t^T]^{-1} \\ &\mathbf{x}_t^t = \mathbf{x}_t^{t-1} + \mathbf{K}_t(\mathbf{y}_t^{obs} - \mathbf{H}_t\mathbf{x}_t^{t-1}), \quad \mathbf{P}_t^t = [\mathbf{I}_n - \mathbf{K}_t\mathbf{H}_t]\mathbf{P}_t^{t-1} \end{aligned} \tag{4.17}$$

Remark 4.1 The gradient-based likelihood techniques implemented using the procedure shown above, suffer from a number of problems. For example, the likelihood function need not increase from iteration to iteration. This happens when $\mathbf{R}^{(i)}$ is indefinite or negative-definite (Gupta and Mehra 1974). On the other hand in EM algorithm the likelihood always increases and convergence to a stationary point is guaranteed (Wu 1983). In order to improve the chances of convergence to a global maximum, a good choice of the initial estimate is critical. Hence, we have chosen to use an efficient subspace identification technique to provide a good initial estimate of the model parameters. It is interesting to note that, while the gradient-based techniques are generally expected to have faster convergence than the EM algorithm, Isaksson (1993) shows through simulation that the EM can be much faster than the gradient-based techniques when there is a significant amount of missing data.

4.4.4 EM algorithm

The EM algorithm is a simple and efficient alternative for MLE from incomplete data records. It is an iterative approach which involves an Expectation step (E-step) and a

Maximization step (M-step). In each iteration, the E-step is used to obtain the expected value of the complete-data likelihood function conditioned on all the available data and the estimated model parameters from the previous iteration. This is followed by the M-step in which a new set of model parameters is obtained by maximizing the E-step likelihood function.

Denote the observed data set as $\mathbf{Z}_N$ and the unobserved data set as $\mathbf{X}_N$. For example, the states can be treated as the unobserved data set. Assume that the *complete* data set consists of both $\mathbf{Z}_N$ and $\mathbf{X}_N$. Then the idea is to maximize the joint probability density function of the observed data, denoted by $f_z(\mathbf{Z}_N|\boldsymbol{\Theta})$, where $\boldsymbol{\Theta}$ is the unknown parameter vector. The distribution of the complete data can be factored as,

$$f_{zx}(\mathbf{Z}_N, \mathbf{X}_N|\boldsymbol{\Theta}) = f_z(\mathbf{Z}_N|\boldsymbol{\Theta}) f_x(\mathbf{X}_N|\mathbf{Z}_N, \boldsymbol{\Theta}) \tag{4.18}$$

Hence, the log-likelihood can be decomposed as,

$$\ell(\boldsymbol{\Theta}|\mathbf{Z}_N, \mathbf{X}_N) = \ell(\boldsymbol{\Theta}|\mathbf{Z}_N) + \log f_x(\mathbf{X}_N|\mathbf{Z}_N, \boldsymbol{\Theta}) \tag{4.19}$$

This can be rewritten as,

$$\ell(\boldsymbol{\Theta}|\mathbf{Z}_N) = \ell(\boldsymbol{\Theta}|\mathbf{Z}_N, \mathbf{X}_N) - \log f_x(\mathbf{X}_N|\mathbf{Z}_N, \boldsymbol{\Theta}) \tag{4.20}$$

The right hand side of Eq. 4.20 depends on the conditional density of the unobserved data, $\mathbf{X}_N$ given $\mathbf{Z}_N$ and $\boldsymbol{\Theta}$, $f_x(\mathbf{X}_N|\mathbf{Z}_N, \boldsymbol{\Theta})$. Consider the expected value of Eq. 4.20 conditioned on the observed data $\mathbf{Z}_N$, and an estimate of the parameters, $\boldsymbol{\Theta}^{k-1}$ obtained from a previous iteration.

$$\ell(\boldsymbol{\Theta}|\mathbf{Z}_N) = Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{k-1}) - H(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{k-1}) \tag{4.21}$$

where,

$$\begin{aligned} Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{k-1}) &\equiv E\left(\log f_{zx}(\mathbf{Z}_N, \mathbf{X}_N|\boldsymbol{\Theta})|\mathbf{Z}_N, \boldsymbol{\Theta}^{k-1}\right) \\ H(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{k-1}) &\equiv E\left(\log f_x(\mathbf{X}_N|\mathbf{Z}_N, \boldsymbol{\Theta})|\mathbf{Z}_N, \boldsymbol{\Theta}^{k-1}\right) \end{aligned} \tag{4.22}$$

Theorem 4.1 Every EM algorithm increases $\ell(\boldsymbol{\Theta}|\mathbf{Z}_N)$ at each iteration, *i.e.*,

$$\ell(\boldsymbol{\Theta}^k|\mathbf{Z}_N) \geq \ell(\boldsymbol{\Theta}^{k-1}|\mathbf{Z}_N) \tag{4.23}$$

with equality if and only if

$$Q(\boldsymbol{\Theta}^k|\boldsymbol{\Theta}^{k-1}) = Q(\boldsymbol{\Theta}^{k-1}|\boldsymbol{\Theta}^{k-1}) \tag{4.24}$$

Proof: This theorem is a key result of Dempster *et al.* (1977). Consider a sequence of iterates, $\ell(\boldsymbol{\Theta}^0), \ell(\boldsymbol{\Theta}^1), \ldots$, where $\ell(\boldsymbol{\Theta}^k) = M(\ell(\boldsymbol{\Theta}^{k-1}))$ for some function $M(.)$. The difference in the values of $\ell(\boldsymbol{\Theta}|\mathbf{Z}_N)$ at successive iterates is,

$$\begin{aligned}\ell(\boldsymbol{\Theta}^k|\mathbf{Z}_N) - \ell(\boldsymbol{\Theta}^{k-1}|\mathbf{Z}_N) =& [Q(\boldsymbol{\Theta}^k|\boldsymbol{\Theta}^{k-1}) - Q(\boldsymbol{\Theta}^{k-1}|\boldsymbol{\Theta}^{k-1})] \\ &- [H(\boldsymbol{\Theta}^k|\boldsymbol{\Theta}^{k-1}) - H(\boldsymbol{\Theta}^{k-1}|\boldsymbol{\Theta}^{k-1})]\end{aligned} \tag{4.25}$$

An EM algorithm chooses $\boldsymbol{\Theta}^k$ to maximize $Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{k-1})$. Hence, the first part of the RHS of Eq. 4.25 is positive. For the second part, we have the following lemma:

Lemma 4.5

$$H(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{k-1}) \leq H(\boldsymbol{\Theta}^{k-1}|\boldsymbol{\Theta}^{k-1}) \tag{4.26}$$

This result is due to Dempster *et al.* (1977) and is proved using Jensen's inequality (Rao (2001), p. 58).

Proof:

Definition 4.1 A function $g(x)$ is *convex* if $\forall\ \alpha, \beta > 0, \alpha + \beta = 1$,

$$g(\alpha x + \beta y) \leq \alpha g(x) + \beta g(y) \tag{4.27}$$

Remark 4.2 If $g(x)$ has a second derivative in $[a, b]$, then a necessary and sufficient condition for it to be convex on that interval is that the second derivative $g''(x) > 0$ for all x in $[a, b]$.

Observation 4.2 For a convex function $g(x)$, at any point x_0, the right and left derivatives, $g'_+(x_0)$ and $g'_-(x_0)$ exist.

Proof: Let $x_0 < x_1 < x_2$. Choose, $\alpha = \dfrac{x_2 - x_1}{x_2 - x_0}$, $\beta = \dfrac{x_1 - x_0}{x_2 - x_0}$, $x = x_0$ and $y = x_2 \Rightarrow \alpha x + \beta y = x_1$.

From Eq. 4.27,

$$\begin{aligned}g(x_1) &\leq \alpha g(x_0) + \beta g(x_2) \\ (x_2 - x_0)g(x_1) &\leq (x_2 - x_1)g(x_0) + (x_1 - x_0)g(x_2)\end{aligned} \tag{4.28}$$

Add $x_0 g(x_0)$ to both sides and rearrange,

$$\frac{g(x_1) - g(x_0)}{x_1 - x_0} \leq \frac{g(x_2) - g(x_0)}{x_2 - x_0} \tag{4.29}$$

This shows that $(g(x) - g(x_0))/(x - x_0)$ decreases as $x \rightarrow x_0$.

Let $x_{-1} < x_0 < x_1$. Choose, $\alpha = \dfrac{x_1 - x_0}{x_1 - x_{-1}}$, $\beta = \dfrac{x_0 - x_{-1}}{x_1 - x_{-1}}$, $x = x_{-1}$ and $y = x_1 \Rightarrow \alpha x + \beta y = x_0$.

From Eq. 4.27,

$$\begin{aligned} g(x_0) &\le \alpha g(x_{-1}) + \beta g(x_1) \\ (x_1 - x_{-1})g(x_0) &\le (x_1 - x_0)g(x_{-1}) + (x_0 - x_{-1})g(x_1) \end{aligned} \tag{4.30}$$

Add $x_0 g(x_0)$ to both sides and rearrange,

$$\frac{g(x_{-1}) - g(x_0)}{x_{-1} - x_0} \le \frac{g(x_1) - g(x_0)}{x_1 - x_0} \tag{4.31}$$

This shows that $(g(x) - g(x_0))/(x - x_0)$ is bounded from below.

Since $(g(x) - g(x_0))/(x - x_0)$ decreases as $x \to x_0$ and is bounded from below, the right derivative $g'_+(x_0)$ exists.

Similarly, $g'_-(x_0)$ exists and $g'_-(x_0) \le g'_+(x_0)$. ■

Observation 4.3 Let K be such that, $g'_-(x_0) \le K \le g'_+(x_0)$. Then for all x,

$$g(x) \ge g(x_0) + K(x - x_0) \tag{4.32}$$

Proof: If $x = x_1 > x_0$,

$$\frac{g(x_1) - g(x_0)}{x_1 - x_0} \ge g'_+(x_0) \ge K \tag{4.33}$$

If $x = x_1 < x_0$,

$$\frac{g(x_1) - g(x_0)}{x_1 - x_0} \le g'_-(x_0) \le K \tag{4.34}$$

■

Observation 4.4 Jensen's Inequality: *If x is a random variable such that $E(x) = \mu$ and $g(x)$ is a convex function, then*

$$E\left[g(x)\right] \ge g\left[E(x)\right] \tag{4.35}$$

with equality if and only if x is a degenerate distribution at μ (A degenerate distribution is one which always has the same value, $f(x) = \mu$).

Proof: Eq. 4.35 can be proved by considering Eq. 4.32 with x_0 replaced by μ and applying the expectation operator.

$$\begin{aligned} g(x) &\ge g(\mu) + K(x - \mu) \\ E[g(x)] &\ge E[g(\mu) + K(x - \mu)] \\ E[g(x)] &\ge g(\mu) + K(E[x] - \mu) \\ E[g(x)] &\ge g(E[x]) \end{aligned} \tag{4.36}$$

■

Remark 4.3 A concave function $g_1(x)$ can be considered as the negative of a convex function $g(x)$. In this case Jensen's inequality can be written as, $E\left[g_1(x)\right] \leq g_1\left[E(x)\right]$.

Remark 4.4 The log function is concave in $(0, \infty)$ because it has a second derivative in $(0, \infty)$ which is always negative.

Eq. 4.26 can be established using,

$$\begin{aligned} H(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{k-1}) &= E\left(\log f_x(\mathbf{X}_N|\mathbf{Z}_N, \boldsymbol{\Theta})|\mathbf{Z}_N, \boldsymbol{\Theta}^{k-1}\right) \\ &\leq \log E\left(f_x(\mathbf{X}_N|\mathbf{Z}_N, \boldsymbol{\Theta})|\mathbf{Z}_N, \boldsymbol{\Theta}^{k-1}\right) \\ &= H(\boldsymbol{\Theta}^{k-1}|\boldsymbol{\Theta}^{k-1}) \end{aligned} \tag{4.37}$$

■

Using this lemma, it has been established that the difference in the H-functions in Eq. 4.25 is negative. Hence, the likelihood of the observed data increases. It leads to the result that $\ell(\boldsymbol{\Theta}^k|\mathbf{Z}_N)$ converges to a stationary value of $\ell(\boldsymbol{\Theta}|\mathbf{Z}_N)$. ■

In effect, the EM procedure reduces to finding the expected value of the Q-function at each iteration conditioned on all the available data and the estimated parameters from the previous iteration, followed by a multivariate regression. Convergence results for the EM algorithm under general conditions are given in Wu (1983). From an application point-of-view, the convergence can be monitored by calculating the negative-log-likelihood function given in Eq. 4.12.

Algorithm 4.2 The EM algorithm can be summarized by the following steps:

- Obtain an initial estimate of the parameter vector, $\boldsymbol{\Theta}^0$.
- Carry out the following steps at each iteration, k, until convergence:
  - **Expectation (E-step)**: Find the expected value of the complete data log likelihood function(Q-function) given the observed data set, $\mathbf{Z}_N$ and the previously estimated parameter vector, $\boldsymbol{\Theta}^{k-1}$. This conditional expectation is obtained using Kalman smoothers which are in turn implemented using Kalman filters and these expressions were provided in Sec. 4.4.2.
  - **Maximization (M-step)**: Maximize the Q-function with respect to the parameter vector. The model parameters estimated in each iteration depend on the observed data and the smoothed state estimates and covariances obtained in the E-step. Expressions for calculating these parameters are provided in Sec. 4.5.1.

The above steps ensure that the log likelihood function of the observed data increases at every iteration. Therefore, the EM algorithm is guaranteed to converge to a local minimum of the likelihood function. However, there are a few drawbacks in using this algorithm. It can be sensitive to the initial guess and also the rate of convergence can be slow. In order to avoid poor initial parameter estimates from irregularly sampled data, identification of an FIR model of the process is suggested.

4.5 Identification of the initial model

In this section we present a realization-based subspace identification technique (Kung 1978, Viberg 1995) which can be used to identify the initial model. Let us represent the process described in Eq. 4.1 as a multivariate FIR model:

$$y_{t,a} = \sum_{b=0}^{m} \sum_{c=0}^{2s-1} h_{abc} u_{t-c,b} + \nu_{t,a}, \quad \forall \ a = 1, \ldots, p \tag{4.38}$$

where, ν represents an arbitrary noise process. $2s-1$ is the number of terms involved in the FIR expansion and should be much larger than n, the number of states expected to be present in Eq. 4.1. The impulse response coefficients in Eq. 4.38 can be estimated using a simple linear regression even in the presence of some missing observations. The variance of the FIR parameters will be large but the quality of the corresponding state space matrices are made better by using a model reduction step which involves singular value decomposition. Following the estimation of the impulse response coefficients $\hat{h}_{abc}$ we can form a set of matrices of the impulse response coefficients, $\hat{H}_r \in \Re^{p \times m}$ corresponding to the lags $r = 0, \ldots, 2s-1$. Using the matrices $\hat{H}_r$ we can form a Hankel matrix $\hat{\mathbf{H}}$.

$$\hat{\mathbf{H}} = \begin{bmatrix} \hat{H}_1 & \hat{H}_2 & \cdots & \hat{H}_s \\ \hat{H}_2 & \hat{H}_3 & \cdots & \hat{H}_{s+1} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{H}_s & \hat{H}_{s+1} & \cdots & \hat{H}_{2s-1} \end{bmatrix} \in \Re^{ps \times ms} \tag{4.39}$$

The matrix $\mathbf{H}$ is the product of the extended observability and controllability matrices, *i.e.*, $\mathbf{H} = \mathbf{\Gamma}_s \mathbf{\Omega}_s$, where $\mathbf{\Gamma}_s$ is the extended observability matrix and $\mathbf{\Omega}_s$ is the extended controllability matrix. Estimates of these matrices can be obtained by performing a singular value decomposition of $\hat{\mathbf{H}}$ (Viberg 1995).

$$\begin{aligned} \hat{\mathbf{H}} &= \hat{\mathbf{Q}}\hat{\mathbf{S}}\hat{\mathbf{V}}^T = \begin{bmatrix} \hat{\mathbf{Q}}_s & \hat{\mathbf{Q}}_n \end{bmatrix} \begin{bmatrix} \hat{\mathbf{S}}_s & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{V}}_s^T \\ \hat{\mathbf{V}}_n^T \end{bmatrix} \\ \hat{\mathbf{\Gamma}}_s &= \hat{\mathbf{Q}}_s \hat{\mathbf{S}}_s^{\frac{1}{2}}, \quad \hat{\mathbf{\Omega}}_s = \hat{\mathbf{S}}_s^{\frac{1}{2}} \hat{\mathbf{V}}_s^T \end{aligned} \tag{4.40}$$

Theoretically, we can choose the appropriate system order by looking at a plot of the singular values. However, in practice the singular values will all be positive and hence it is difficult to decide on the exact number of zero singular values. In the examples used to illustrate these ideas, we make an appropriate choice of the model order so that the neglected singular values only reflect the noise variance, since the goal here is not to perform analysis with biased, reduced complexity models.

The $\hat{\mathbf{B}}$ and $\hat{\mathbf{C}}$ matrices can be read out from the first block column of $\hat{\mathbf{\Omega}}_s$ and the first block row of $\hat{\mathbf{\Gamma}}_s$ respectively. $\hat{\mathbf{A}}$ can be estimated from the shift invariant structure of either $\hat{\mathbf{\Gamma}}_s$ or $\hat{\mathbf{\Omega}}_s$ (Overschee and DeMoor 1996).

Remark 4.5 Using an FIR model structure for the initial model restricts the model to the output-error class of models. Hence the identification result yields only the deterministic sub-system (*i.e.*, the plant dynamics only). Any additional stochastic states are ignored. In order to include these, it is necessary to fit a pre-whitening filter to the residuals, lump the deterministic and stochastic states and then perform a model reduction step. An optimal method to handle this stochastic realization problem is provided by the EM algorithm because we are dealing with residuals which are not regularly spaced in time. The EM algorithm to deal with this identification problem can be found in Shumway and Stoffer (2000).

4.5.1 Expressions for new model parameters in the M-step

Lemma 4.6 The new model parameters at the end of each iteration are given by,

$$\boldsymbol{\mu}_0 = \mathbf{x}_0^N, \quad \mathbf{\Sigma}_0 = \mathbf{P}_0^N, \quad [\mathbf{A} \mid \mathbf{B}] = \beta_2\beta_3^{-1}, \quad \mathbf{C} = \beta_5\beta_1^{-1},$$
$$\mathbf{R}_w = \tfrac{1}{N}(\beta_1 - \beta_2\beta_3^{-1}\beta_2^T), \quad \mathbf{R}_v = \tfrac{1}{N}(\beta_4 - \beta_5\beta_1^{-1}\beta_5^T).$$

where, the quantities $\beta_1, \ldots, \beta_5$ are defined in the proof.

Proof: The new model parameters at the end of each iteration are obtained by solving the optimization problem in the M-step, which can be written as:

$$\begin{aligned}
\mathbf{\Theta}^k &= \arg\min_{\mathbf{\Theta}}\{-2Q(\mathbf{\Theta}, \mathbf{\Theta}^{k-1})\} \\
&= \arg\min_{\mathbf{\Theta}}\{E(N\log|\mathbf{R}_v| + \text{Syt}|\mathbf{Z}_N, \mathbf{\Theta}^{k-1}) \\
&\qquad + E(\log|\mathbf{\Sigma}_0| + (\mathbf{x}_0 - \boldsymbol{\mu}_0)^T\mathbf{\Sigma}_0^{-1}(\mathbf{x}_0 - \boldsymbol{\mu}_0)|\mathbf{Z}_N, \mathbf{\Theta}^{k-1}) \\
&\qquad + E(N\log|\mathbf{R}_w| + \text{Sxt}|\mathbf{Z}_N, \mathbf{\Theta}^{k-1})\} \\
&= (T_1 + T_2 + T_3 + T_4 + T_5 + T_6)
\end{aligned} \tag{4.41}$$

where,

$$
\begin{aligned}
\text{Sxt} &= \sum_{t=1}^{N}(\mathbf{x}_t - \mathbf{A}\mathbf{x}_{t-1} - \mathbf{B}\mathbf{u}_{t-1})^T\mathbf{R}_w^{-1}(\mathbf{x}_t - \mathbf{A}\mathbf{x}_{t-1} - \mathbf{B}\mathbf{u}_{t-1}) \\
\text{Syt} &= \sum_{t=1}^{N}(\mathbf{y}_t - \mathbf{C}\mathbf{x}_t)^T\mathbf{R}_v^{-1}(\mathbf{y}_t - \mathbf{C}\mathbf{x}_t) \\
T_1 &= \log|\boldsymbol{\Sigma}_0|, \quad T_2 = N\log|\mathbf{R}_w|, \quad T_3 = N\log|\mathbf{R}_v| \\
T_4 &= tr\{\boldsymbol{\Sigma}_0^{-1}\left(\mathbf{P}_0^N + (\mathbf{x}_0^N - \boldsymbol{\mu}_0)(\mathbf{x}_0^N - \boldsymbol{\mu}_0)^T\right)\} \\
T_5 &= tr\{\mathbf{R}_w^{-1}(\beta_1 - \beta_2[\mathbf{A} \mid \mathbf{B}]^T - [\mathbf{A} \mid \mathbf{B}]\beta_2^T + [\mathbf{A} \mid \mathbf{B}]\beta_3[\mathbf{A} \mid \mathbf{B}]^T)\} \\
T_6 &= tr\{\mathbf{R}_v^{-1}\left(\beta_4 - \beta_5\mathbf{C}^T - \mathbf{C}\beta_5^T + \mathbf{C}\beta_1\mathbf{C}^T\right)\}
\end{aligned}
\tag{4.42}
$$

β_1 to β_5 are functions of the observed data and smoothed estimates.

$$
\begin{aligned}
\beta_1 &= \sum_{t=1}^{N}(\mathbf{P}_t^N + \mathbf{x}_t^N(\mathbf{x}_t^N)^T) \\
\beta_2 &= \left[\begin{array}{cc} \sum_{t=1}^{N}(\mathbf{P}_{t,t-1}^N + \mathbf{x}_t^N(\mathbf{x}_{t-1}^N)^T) & \sum_{t=1}^{N}(\mathbf{x}_t^N\mathbf{u}_{t-1}^T) \end{array}\right] \\
\beta_3 &= \left[\begin{array}{cc} \sum_{t=1}^{N}(\mathbf{P}_{t-1}^N + \mathbf{x}_{t-1}^N(\mathbf{x}_{t-1}^N)^T) & \sum_{t=1}^{N}(\mathbf{x}_{t-1}^N\mathbf{u}_{t-1}^T) \\ \sum_{t=1}^{N}(\mathbf{x}_{t-1}^N\mathbf{u}_{t-1}^T)^T & \sum_{t=1}^{N}\mathbf{u}_{t-1}\mathbf{u}_{t-1}^T \end{array}\right] \\
\beta_4 &= \sum_{t=1}^{N}(\mathbf{y}_t\mathbf{y}_t^T) \quad \beta_5 = \sum_{t=1}^{N}(\mathbf{y}_t(\mathbf{x}_t^N)^T)
\end{aligned}
\tag{4.43}
$$

The solution we obtain by setting the first derivatives of $-2Q(\Theta, \Theta^{k-1})$ to zero is given by: $\boldsymbol{\mu}_0 = \mathbf{x}_0^N$, $\boldsymbol{\Sigma}_0 = \mathbf{P}_0^N$, $[\mathbf{A} \mid \mathbf{B}] = \beta_2\beta_3^{-1}$, $\mathbf{C} = \beta_5\beta_1^{-1}$,
$\mathbf{R}_w = \frac{1}{N}(\beta_1 - \beta_2\beta_3^{-1}\beta_2^T)$, $\mathbf{R}_v = \frac{1}{N}(\beta_4 - \beta_5\beta_1^{-1}\beta_5^T)$. ■

4.6 Handling missing observations in the EM-based identification

The main strength of the EM algorithm lies in its ability to handle missing observations in the identification data set. It arises from the provision for including unobserved data which is inherent in the EM algorithm formulation. The Missing-data Kalman filter expressions (Sec. 4.4.3) can be used in the missing data case.The Kalman smoother can be used with the predicted and filtered state estimates and covariances obtained from the modified Kalman filter. In addition to these changes, the expressions for estimating the C and $\mathbf{R}_v$ matrices in the M-step also change. In the M-step of the EM algorithm, the smoothed state and covariances from the modified Kalman smoother are used. In addition the expression for

T_6 changes, affecting the expressions for $\mathbf{C}$ and $\mathbf{R}_v$. Assume that there are N_m instances in which $\mathbf{y}_2$ is not measured and $N_o = N - N_m$ observations in which $\mathbf{y}_2$ is measured.

$$\begin{aligned} T_6 &= E\left((\mathbf{y}_t - \mathbf{C}\mathbf{x}_t)(\mathbf{y}_t - \mathbf{C}\mathbf{x}_t)^T | \{\mathbf{Y}_N^{(1)}, \mathbf{U}_N\}, \mathbf{\Theta}^{k-1}\right) \\ &= \sum_{N_o \text{cases}} ((\mathbf{y}_t - \mathbf{C}\mathbf{x}_t^N)(\mathbf{y}_t - \mathbf{C}\mathbf{x}_t^N)^T + \mathbf{C}\mathbf{P}_t^N\mathbf{C}^T) + \sum_{N_m \text{cases}} \begin{bmatrix} S_{ym} & 0 \\ 0 & \mathbf{R}_{v2} \end{bmatrix} \end{aligned} \tag{4.44}$$

where, $S_{ym} = (\mathbf{y}_{1,t} - \mathbf{C}_1\mathbf{x}_t^N)(\mathbf{y}_{1,t} - \mathbf{C}_1\mathbf{x}_t^N)^T + \mathbf{C}_1\mathbf{P}_t^N\mathbf{C}_1^T$ and $\mathbf{Y}_N^{(1)}$ is the observed data set defined according to Eq. 4.16.

4.7 Illustrative applications

In this section we present examples to illustrate the application of the proposed identification technique. First, we present an example of a simplified version of the EM procedure. Following this, we present the application of the EM algorithm to a computer-simulated underdamped system, a pilot-scale Continuous-flow Stirred Tank Heater (CSTH) process present in the Computer Process Control laboratory at the University of Alberta and to the bleaching unit of Millar-Western's Bleached-Chemi Thermo-Mechanical Pulp (BCTMP) mill in Whitecourt, Alberta.

4.7.1 Example: Simplified version of EM procedure

We now present an example illustrating the use of a simplified version of the EM algorithm in estimating models from multi-rate data.

Consider an ARX model,

$$y(k) = 0.8y(k-1) + 0.3u(k-1) + e(k) \tag{4.45}$$

where $e(k)$ is normally distributed white noise with variance $\sigma_e^2 = 0.01$. Assume that the input is sampled regularly, the output is available at every alternate sampling instant and that $y(1)$ is known. Then the following objective function based on squared prediction errors can be used for identifying the model parameters:

$$V_N(\theta) := \frac{1}{N}\sum_{k=1}^{N} \varepsilon(t,\theta)^2 = \frac{1}{N}\sum_{k=1}^{N} [y(k) - \theta_2 y(k-1) - \theta_1 u(k-1)]^2 \tag{4.46}$$

where N is the data length and $\theta = [\theta_1 \quad \theta_2]^T$. Since only alternate data points are available, the above objective function cannot be evaluated. Instead, it is possible to estimate the expected value of the above objective function given the estimate of θ from the previous iteration, $\hat{\theta}^{(j-1)}$ *i.e.*,

$$E\left[V_N(\theta)|\hat{\theta}^{(j-1)}, Z_N\right] = E[\frac{1}{N}\sum_{k=1}^{N}[y(k) - \theta_2 y(k-1) - \theta_1 u(k-1)]^2 \,|\hat{\theta}^{(j-1)}, Z_N] \tag{4.47}$$

where Z_N denotes all the available data. Now two cases can be considered:
Case I: $y(k)$ is known:

$$\begin{aligned} E\,[y(k) - \theta_2 y(k-1) - \theta_1 u(k-1)]^2 = {} & (y(k) - \theta_1 u(k-1))^2 + \theta_2^2(\hat{\theta}_1^{(j-1)} u(k-2) \\ & + \hat{\theta}_2^{(j-1)} y(k-2))^2 + \theta_2^2 \sigma_e^2 \\ & - 2(y(k) - \theta_1 u(k-1))\theta_2(\hat{\theta}_1^{(j-1)} u(k-2) \\ & + \hat{\theta}_2^{(j-1)} y(k-2)) \end{aligned} \tag{4.48}$$

Case II: $y(k)$ is unknown:

$$\begin{aligned} E\,[y(k) - \theta_2 y(k-1) - \theta_1 u(k-1)]^2 = {} & (\hat{\theta}_1^{(j-1)} u(k-1) + \hat{\theta}_2^{(j-1)} y(k-1))^2 \\ & + \sigma_e^2 + (\theta_1 u(k-1) + \theta_2 y(k-1))^2 \\ & - 2(\theta_1 u(k-1) + \theta_2 y(k-1))(\hat{\theta}_1^{(j-1)} u(k-1) \\ & + \hat{\theta}_2^{(j-1)} y(k-1)) \end{aligned} \tag{4.49}$$

Using Eqs. 4.48 and 4.49 in Eq. 4.47 it is possible to find the model parameters at the current (j^{th}) iteration:

$$\theta^{(j)} = \min_{\theta} E\left[V_N(\theta)|\hat{\theta}^{(j-1)}, Z_N\right] \tag{4.50}$$

The iterations are performed until the parameters converge. A plot showing the two parameters in this example and the number of iterations is shown in Fig. 4.1. The corresponding decrease in the average prediction error as a function of the iterations is shown in Fig. 4.2.

The estimated model parameters converge to the true parameters in spite of the missing data. In general, the estimates using EM algorithm need not converge to the true parameters when the number of samples is finite. However, the estimated parameters converge to the true parameters asymptotically. On the other hand, the parameters of the least squares model obtained by interpolating the data are $\hat{\theta}_1 = 0.83$ and $\hat{\theta}_2 = 0.24$. These parameters are clearly biased. In general, the estimated models are biased if arbitrary interpolation methods are used to fill the missing data points.

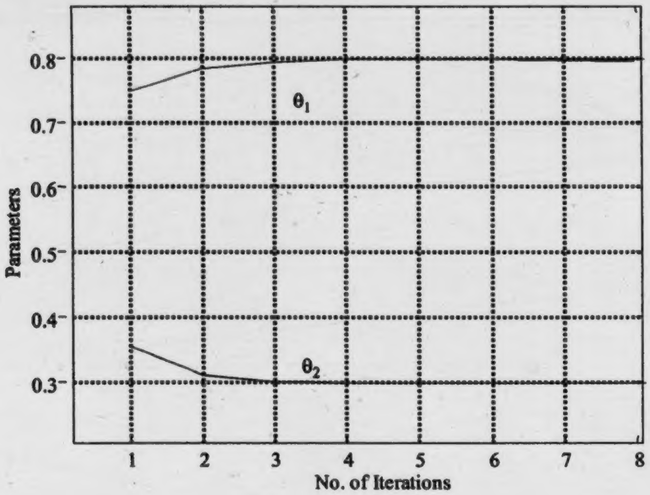


Figure 4.1: Plot of θ_1 and θ_2 as a function of number of iterations

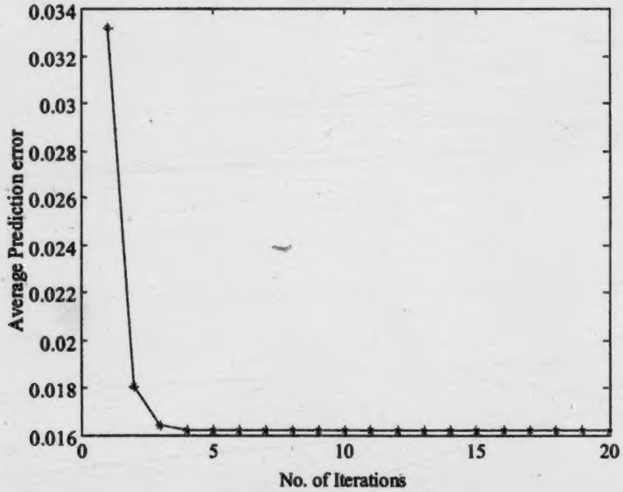


Figure 4.2: Plot of the average prediction error as a function of number of iterations

4.7.2 Simulated Case-study: 3^{rd} order underdamped system

In this example, we use the proposed EM-based strategy for identifying the underdamped system defined by the state-space matrices:

$$A = \begin{bmatrix} 0.3688 & 0.4767 & 0.0114 \\ -0.5976 & 0.6095 & -0.5408 \\ -0.0156 & -0.0686 & 0.0422 \end{bmatrix} \quad B = \begin{bmatrix} 0.34 \\ 0.56 \\ 0.78 \end{bmatrix} \quad C = \begin{bmatrix} 1.2 & 0.96 & 1.5 \end{bmatrix}$$

$$R_w = \begin{bmatrix} 0.0407 & 0.0001 & 0.0015 \\ 0.0001 & 0.0407 & -0.0020 \\ 0.0015 & -0.0020 & 0.0428 \end{bmatrix} \quad R_v = 0.0398$$

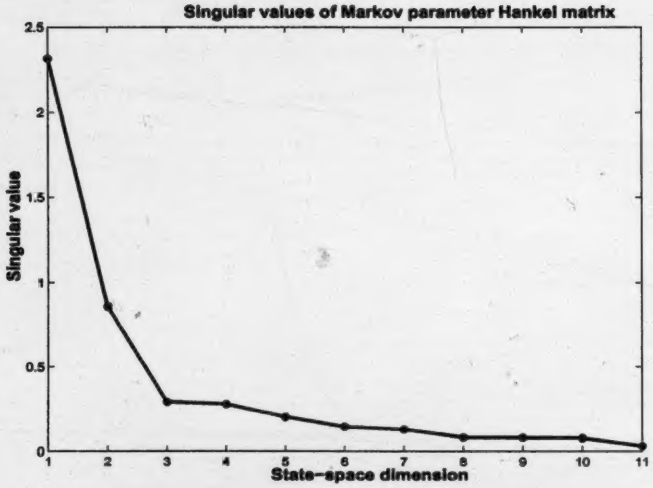


Figure 4.3: Singular value plot for order selection

The system is excited using $N = 5000$ samples of a random binary input signal generated by the Matlab© command *idinput*. To study the performance of the EM algorithm, every 3^{rd} and 5^{th} output sample was dropped. Hence 2333 samples out of the 5000 samples were dropped. The FIR model was identified with 20 FIR coefficients. Note that this choice of the number of coefficients, requires some knowledge of the settling times in the system. The singular value plot was used for selecting the order of the initial model (Fig. 4.3). It clearly shows that the desired order is $n = 3$. Following the identification of the FIR-based initial model, the EM-algorithm iterations were started and the Negative Log Likelihood (NLL) function values were displayed for monitoring convergence. A plot of these values, shown in Fig. 4.4 indicates that the NLL values have satisfactorily converged in about 30 iterations. A comparison of the step responses of the true system and the identified model (Fig. 4.5) shows that the identified model is close to the true system.

In addition, a *zoomed-in* plot with the true output values, measured observations and the

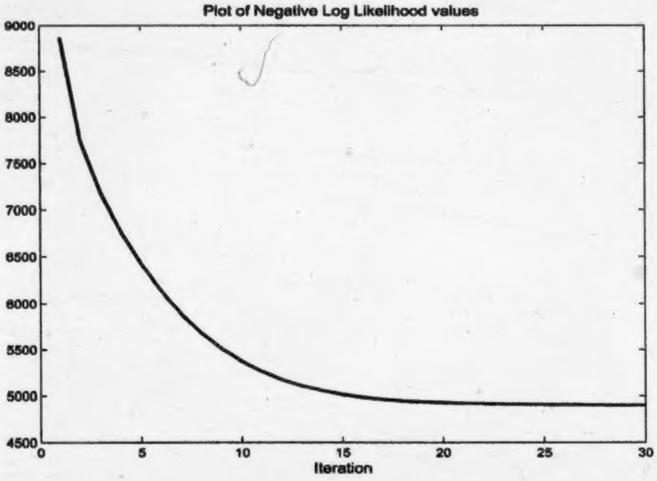


Figure 4.4: Monitoring convergence using the Negative log likelihood function

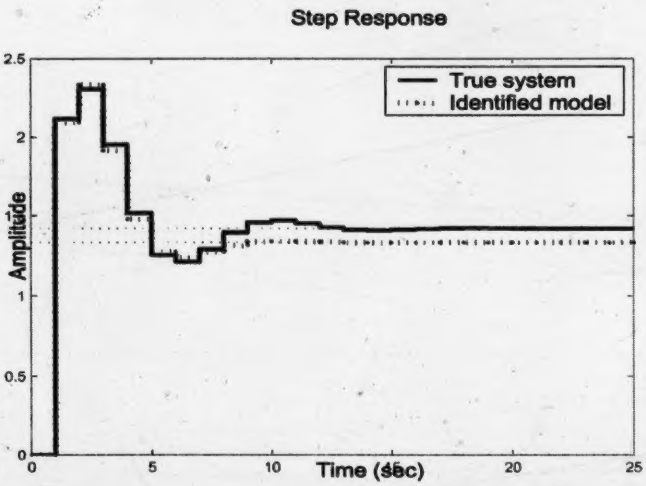


Figure 4.5: Comparison of step responses of the underdamped system

Kalman filter predictions is shown in Fig. 4.6. It is clear that the Kalman filter predictions track the true output satisfactorily. It is also evident that linear interpolation between observed samples would have failed to reconstruct the true output satisfactorily.

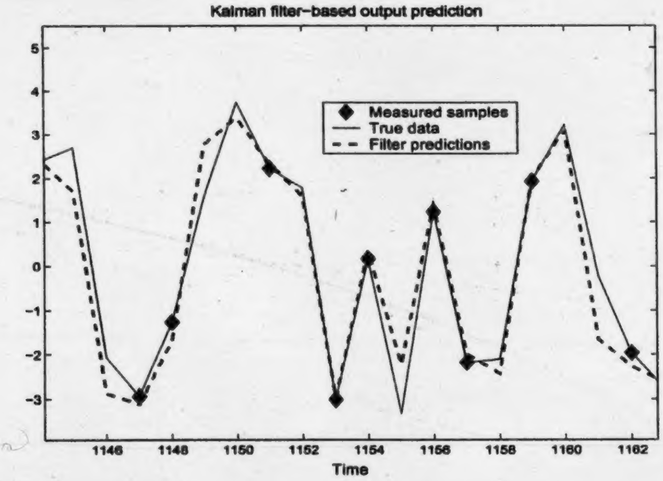


Figure 4.6: Comparison of Kalman filter predictions with true output in the underdamped system

4.7.3 Laboratory Case-study: Continuous-flow Stirred Tank Heater (CSTH) process

We apply the proposed algorithm for the identification of linear time-invariant models for a pilot-scale Continuous-flow Stirred Tank Heater (CSTH) process. A schematic of this laboratory process is shown in Fig. 4.7.

The system has two manipulated inputs, cold water valve position (U_1) and the Steam valve position(U_2). The measured outputs are the water level in the tank (Y_1) and the temperature of water as it exits from the tank (Y_2). The inlet valve positions, level and exit temperature are measured on a scale of $4 - 20mA$. The system is excited using $N = 5000$ samples of random binary input signals generated by the Matlab© *idinput* command with the input frequency range, 0 to 0.2, expressed as a fraction of the Nyquist frequency. The complete data set was first used to identify a 3^{rd} order state-space model using the PEM function in Matlab©, for comparative purposes (since we do not have perfect knowledge of the true system step responses). Following this, every 2^{nd} and 3^{rd} temperature observation was dropped to study the performance of the EM algorithm. Hence, 3333 samples out of the 5000 samples were dropped. In addition, the samples between $t = 1000$ and $t = 1200$ were dropped. This was done to verify the quality of the Kalman filter predictions in the absence of measurements over a period of time. Hence, 3400 samples out of the 5000 samples

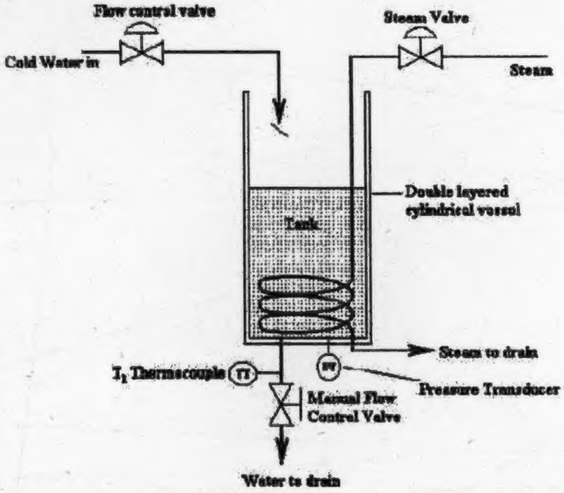

Flow control valve
Cold Water in
Steam Valve
Steam
Tank
Double layered cylindrical vessel
Steam to drain
T1 Thermocouple
TT
PT
Pressure Transducer
Manual Flow Control Valve
Water to drain

Figure 4.7: Schematic of a pilot-scale CSTH process

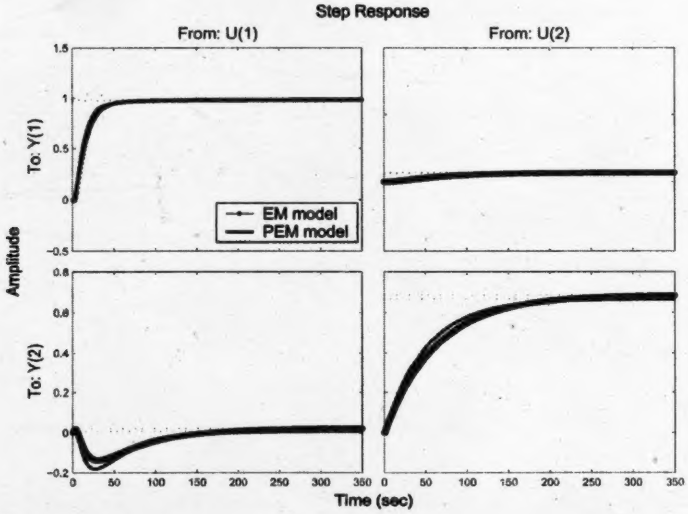


Figure 4.8: Comparison of PEM and EM step responses

were *missing*. In Fig. 4.8 we compare the step responses obtained after 40 iterations of the EM algorithm (with missing data) with the step responses obtained from PEM (with

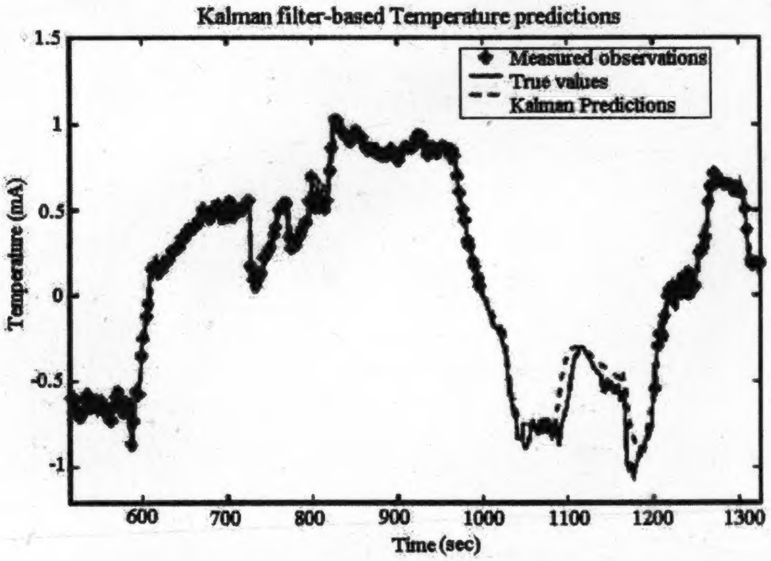


Figure 4.9: Comparison of Kalman filter predictions with true output

no missing data). It is clear that the performance of the EM algorithm is very good in spite of the missing samples. In addition, we present the Kalman filter-based temperature predictions with the true data and the measured values in Fig. 4.9. In this plot, there are no observations made between $t = 1000$ and $t = 1200$. However, it is clear that the Kalman filter predictions continue to remain close to the true output value. This is due to the intelligent usage of the available information in the Kalman filter-based predictor. On the other hand it is clear that if we had used linear interpolation, the predictions would have been erroneous.

4.7.4 Industrial Case-study: Bleaching unit in a BCTMP mill

Finally, we present the results of an industrial application of the proposed identification strategy. In this application, identification of dynamic models was performed for the bleaching operation of a Bleached-Chemi Thermo-Mechanical Pulp process at Millar Western, Whitecourt, AB, Canada. A short description is provided here while the details of the process, complexity of the modeling problem, the gray-box modeling procedure and the implementation are given in the next chapter.

The bleaching unit consists of two towers, has four manipulated inputs and two measured

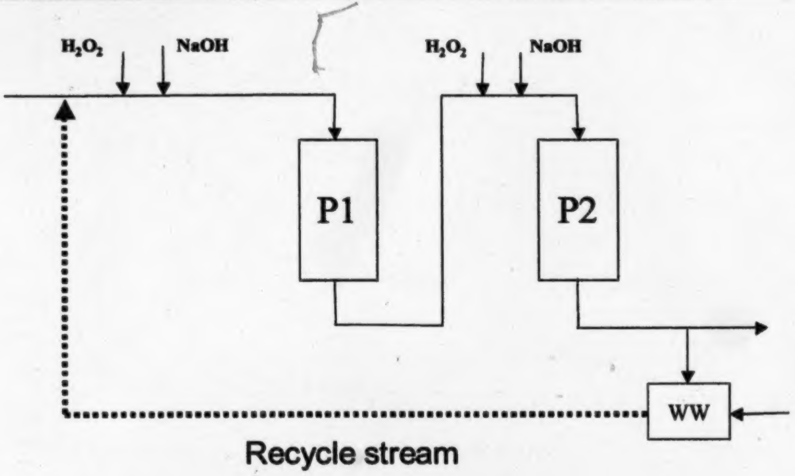


Figure 4.10: Simplified flow sheet of Mechanical Pulp mill

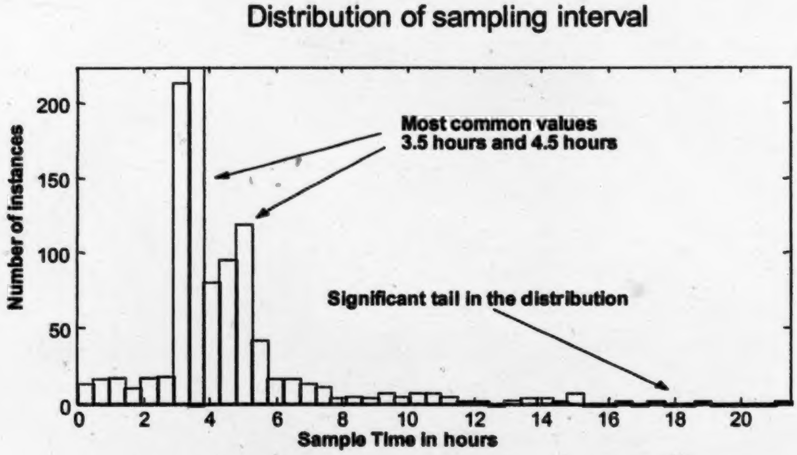


Figure 4.11: Distribution of the output sampling intervals

disturbances. We show the results for one of the quality variables, the pulp brightness. A simplified process schematic is presented in Fig. 4.10. The manipulated inputs were chemical add-rates (Peroxide and Caustic) to the two towers. There were two additional measurements (Aspen and Freeness) which are classified as measured disturbances. The process has plug-flow characteristics (forward path dynamics can be captured by low-order delay dominant models) with significant chemical recycle (average fraction of chemical being recycled must be captured in model). The presence of the recycle stream alters the dynamics of the process significantly. Hence it is necessary to take the recycle effect into account, by using input terms lagged by the delay in the loop.

The distribution of the output sampling intervals, i.e., the time interval between consecutive samples, is shown in Fig. 4.11. It is clear from the histogram that the time interval between consecutive samples is a highly varying quantity. The most common values for this time interval are 3.5 or 4.5 hours. However, there are instances when this interval can take values as large as 20 hours.

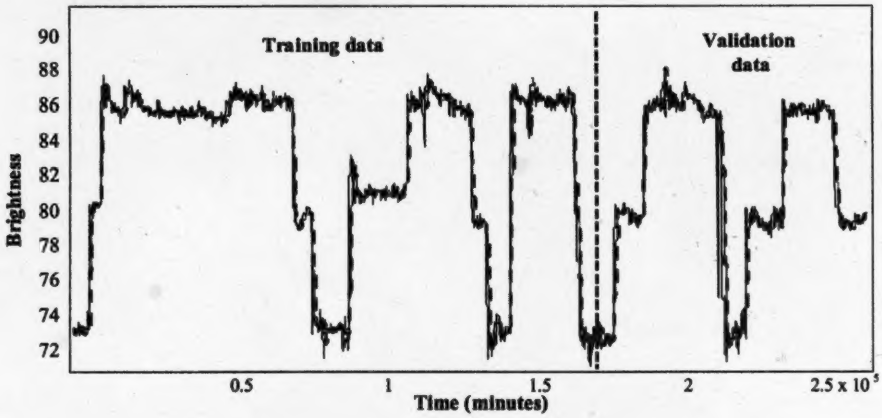


Figure 4.12: Predictions using linear interpolation look good

In addition, we had to use routine operating data for model identification. The main reason for using routine operating data was the prohibitive economic consequences of the loss of productivity accompanying the down-time caused by plant tests. However, it was feasible to use operating data in this case, because it contained enough excitation and a high signal-to-noise ratio.

The process models obtained in this identification exercise have to conform to what we know about the process from physical knowledge. This qualitative knowledge can be summarized as, fast forward-path dynamics, significant recycle and positive gains. When we perform identification after linear interpolation, Fig. 4.12 shows the predictions based on linear interpolation which look very good (*Note that we have masked the output axis for confidentiality reasons*). The correlation coefficient (CC) between the predicted and measured values is 0.97 and the root-mean-squared error (RMSE) value is 1.05. However,

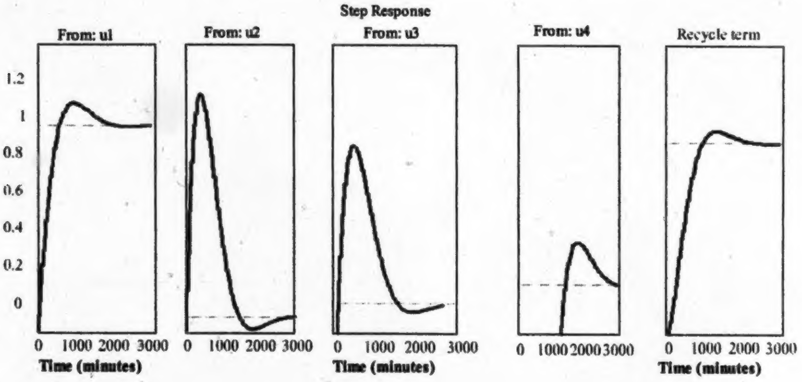


Figure 4.13: Step responses using interpolated data are not satisfactory

the step responses (Fig. 4.13) are very different from the expected step responses. This clearly demonstrates the danger of arbitrary interpolation.

The predictions from the proposed identification strategy are shown in Fig. 4.14 and the corresponding step responses are shown in Fig. 4.15 (*We have masked the output axis for confidentiality reasons*). The correlation coefficient (CC) between the predicted and measured values is 0.984 and the root-mean-squared error (RMSE) value is 0.87. The predictions look good. More importantly, the step responses conform to our qualitative knowledge about the process dynamics.

4.8 Conclusions

An EM-based strategy for identification of chemical processes with irregularly sampled outputs has been presented. The initial model required for the EM algorithm is obtained from FIR coefficients through an SVD procedure. In addition, optimal state estimation and prediction are performed using a Kalman predictor-corrector mechanism which is constructed during the identification. Applications of the proposed approach to simulation, laboratory and industrial case-studies have been presented and these show that the EM-based identification strategy is useful for data-based identification of state-space models even when some output observations are missing.

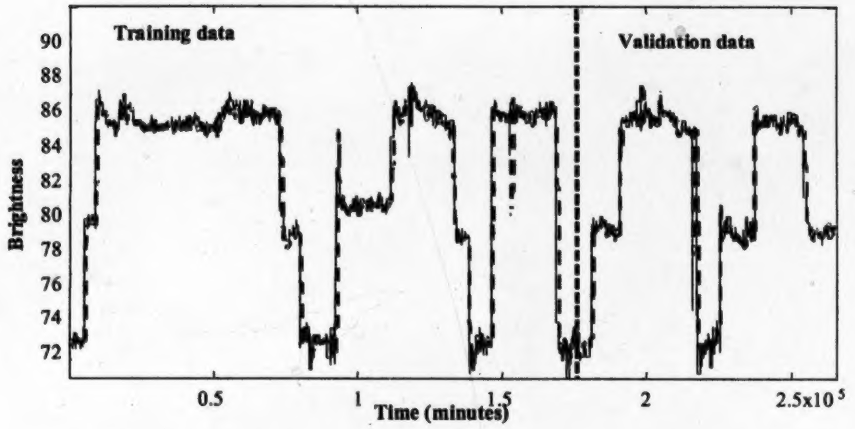


Figure 4.14: Brightness predictions using EM-based strategy

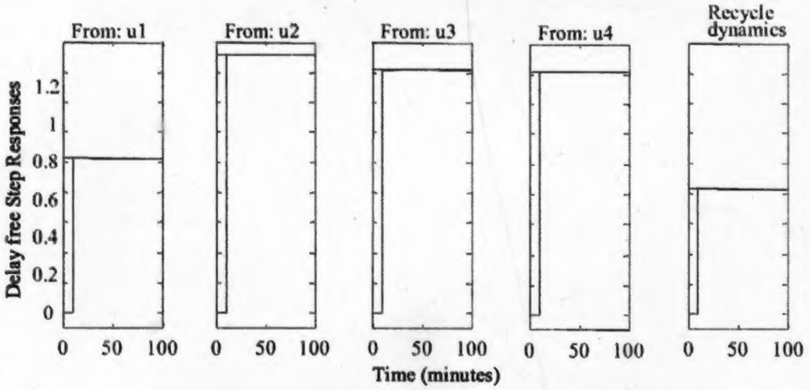


Figure 4.15: Step responses using EM-based strategy

5

Gray-box model identification in a pulp mill

5.1 Overview

The development and application of gray-box identification techniques for modeling the bleaching operation in a Bleached Chemi-Thermo Mechanical Pulp (BCTMP) mill is explained. The process is characterized as a delay dominant recycle process with significant input nonlinearities. The identification was carried out using routine operating data in which the outputs were measured irregularly. The effects of these characteristics and consequent modifications of the system identification techniques are discussed. The resulting models are being used for online prediction and model-based controller design at the mill with satisfactory performance[1].

---

[1] Sections of this chapter have been presented or published as:

1. Raghavan, H., R.B. Gopaluni, S.L. Shah, J. Pakpahan, R.S. Patwardhan and C. Robson, "Gray-box identification of dynamic models for the Bleaching operation in a BCTMP mill", Submitted to Journal of Process Control, Dec 2003.
2. Raghavan, H., S.L. Shah, R.S. Patwardhan, J. Pakpahan, G. Sedgwick and C. Robson, "Identification of delay-dominant recycle systems", Presented in CSChE Annual Conference, Vancouver, Canada, Oct, 2002.
3. Gopaluni R.B., H. Raghavan, R.S. Patwardhan and S.L. Shah, "On the Identification of Consistent Models for Processes with Material & Energy Recycle", Presented in AIChE Annual Conference, San Francisco, USA, Nov, 2003.

5.2 Introduction

Applications such as advanced control, process monitoring and dynamic simulation of chemical processes require the use of dynamic models. Phenomenological models developed using physical principles such as mass and energy conservation are accurate in representing chemical processes. However, the development of such models is a non-trivial exercise. Understanding the intricacies of a chemical process requires a significant investment of time and money. In the mean time simple, data-based empirical models are required to carry-out the day-to-day operations.

Empirical models developed using system identification routines are used to capture the average behavior of the process. Typically, these models treat units of the chemical process as time-invariant, lumped parameter systems subject to temporal variations in the measured inputs and unmeasured disturbances. However, they can be more sophisticated if such complexity is necessary to capture the observed dynamics of the process. Such empirical models have been widely used to control the system at the desired operating point and achieve product quality targets in a large number of chemical processes. Generally, these models are built using batches of measured data collected during routine operation or by conducting specifically designed experiments. Among the traditional methods of dynamic system identification, Prediction Error Methods (PEM) (Ljung 1999), Instrumental variable methods (Söderström and P. Stoica 1983) and Subspace Identification methods (Overschee and DeMoor 1996) are popular.

However it is difficult to apply these techniques directly for the identification of models in an industrial setting for a number of reasons. This is evident in the problem under consideration, *i.e.*, the problem of identifying a sensible model for the bleaching operation in a BCTMP mill from routine operating data. This process can be characterized as a delay-dominant process with significant chemical recycle, irregularly sampled outputs, significant input nonlinearities. The task is difficult because the complexity of the identification task requires the application of ideas used in the solution of a number of challenging problems, some of which have not been addressed satisfactorily in traditional system identification literature.

Commonly used identification routines assume that the data is available at uniformly spaced sample instants. In many chemical processes, variables are measured at differing sampling intervals. This may be in part, due to physical and economic constraints in measuring certain quality variables, such as brightness of pulp. The problem of identifying optimal models when some of the variables are irregularly sampled has been studied in statistical literature using the Expectation Maximization approach (Dempster *et al.* 1977, Shumway and Stoffer 1982, Ninness and Gibson 2002, Gibson and Ninness 2000). The EM approach has also been used for dynamic data rectification using its state estimation properties, in the chemical engineering literature (Singhal and Seborg 2000). An overview of the use of various techniques for identifying ARX models subject to missing data can be found in Isaksson (1993) (Isaksson 1993). In addition sub-optimal techniques using approaches like linear interpolation (Amirthalingam *et al.* 2000) have been proposed in chemical engineering literature. Some continuous-time model identification techniques (Sinha and Rao 1991), which use simple numerical integration procedures like trapezoidal and Simpson's rules to approximate the inter-sample behavior of the process can also be considered as interpolation techniques. There has also been some interest recently in using the lifting operator (Kranc 1957, Freidland 1961, Khargonekar *et al.* 1985, Chen and Francis 1995) to convert the multi-rate identification problem into a slow, single-rate identification problem (Li *et al.* 2001*a*). However, performing unconstrained lifted system identification using subspace identification techniques leads to sub-optimal models in the sense of maximum likelihood estimation. In addition, extracting the fast-rate model and ensuring that these models are causal is difficult. Constrained identification of lifted systems using gray-box identification tools is also being pursued (Wang *et al.* 2004). However, the final model obtained using these techniques has a strong dependence on the initial guess and it is difficult to ensure that these models converge to the global optimum because of the nonlinearity of the optimization problem. The identification problem discussed here was solved using a two-stage procedure. In the first stage, simple FIR-type models were identified using constrained optimization techniques on account of irregular sampling, fast dynamics, large time delays and chemical recycle. Following this, the EM algorithm was applied with the FIR model as the initial guess to obtain maximum likelihood estimates of the model parameters.

Chemical processes operating with material and energy recycle have attracted a lot of attention because of the complexity of their dynamics and the consequent challenges they pose for controller design (Luyben 1994). Research on the empirical model identification, dynamics and control of recycle processes is of great practical importance because of the wide use of these systems in the process industry. While most of the published work on recycle processes concentrates on dynamics and control, there have been relatively few publications which address the corresponding identification problems (Kwok *et al.* 2001, Lakshminarayanan and Takada 2001). When identifying models for delay-dominant recycle processes the complexity of the dynamics should be taken into account. For instance, these systems have staircase-shaped step responses. Hence the use of black-box techniques may lead to the identification of inconsistent models which fail to capture these characteristic staircase-shaped step-responses. Considering these arguments, an approach which guarantees consistent model identification in delay-dominant recycle systems was used for the application under consideration.

Chemical processes often exhibit nonlinear behavior. When the process operates around a small region of a fixed operating point, it can be approximated accurately using a linear model. When this is not the case, it might be necessary to consider nonlinear effects. The term *nonlinear* is modest in that, it tells us what properties the system lacks instead of revealing the properties which the system possesses. Any system that cannot be adequately represented using a linear model can be considered as a nonlinear system. Consequently a large number of model structures have been proposed to characterize nonlinear systems. In addition to simple series expansion models, nonlinear black-box structures can be identified using approximators such as neural networks. A number of articles have been published in engineering literature which give a process control oriented introduction to nonlinear model identification (Pearson and Ogunnaike 1996, Ljung 1999). One of the problems with nonlinear identification is the selection of the appropriate model structure. There are a number of ways in which the regressor can be parameterized. However, it is recommended by experts in the field of model identification (Ljung 1999) that it is better to utilize physical insight into the character of possible nonlinearities for constructing suitable model structures. Hence, it is better to use gray-box structures developed using process knowledge instead of using nonlinear black-box structures because these give the user more confidence in the model and a better insight into the process. Based on process knowledge derived from operational experience, a truncated second-order volterra series model structure is used in the initial model for the BCTMP bleaching operation.

The theory of identification of linear dynamic models from input-output data recommends the use of inputs which are *persistently exciting* (Ljung 1999) up to the desired model order. These specially designed inputs are applied to the plant during dynamic plant tests, during which the productivity of the plant is greatly affected. In practice however, the use of such input signals is avoided in the chemical process industry and in many cases low order dynamic models identified through simple step tests or *bump tests* are used to avoid degradation in the process equipment and loss of productivity accompanying these plant tests. The loss of productivity is a particularly significant factor for plants which have slow dynamics. For example, it may be worthwhile noting that settling times for the bleaching operation of the BCTMP process can be of the order of 24 hours. The use of traditional plant tests for identifying models for such plants would require weeks of dynamic testing which would be prohibitively expensive. The costs of dynamic testing will have to be balanced by the savings which are brought in by the implementation of the advanced control scheme. Hence, dynamic plant testing will have to be justified using long term economic benefit forecasting which is not straightforward. In view of these problems, the use of routine operating data which contains enough excitation through deliberate movement of the manipulated variables (*during grade changes etc.*) for identifying low order models becomes significant. While identifying models from routine operating data, ensuring that these models conform to what we know about the process in terms of gain directions and values is very important. This is because, these models could easily reflect the moves made to the manipulated variables by the operator in response to changes in the output caused by unmeasured disturbances. In the current application, we have used constrained optimization and gray-box identification routines to ensure that the models obtained have correct gain directions.

In the subsequent portions of this chapter, the identification of time-invariant dynamic models for the bleaching operation in a BCTMP mill is discussed. The procedures adopted to solve some of the problems encountered during this identification exercise and some of the issues which are relevant to the online application of the identified models as dynamic output predictors are described. The rest of the chapter is organized as follows: The BCTMP bleaching operation and the corresponding identification problem is described in Sec. 5.3. Following this, the steps taken to solve the problems peculiar to this application are described in Secs. 5.4, 5.5 and 5.6. This is followed by a summary of the results in Sec. 5.7 and concluding remarks in Sec. 5.8.

5.3 Process Description

In this application, gray-box identification of dynamic models for the bleaching operation in a BCTMP process at Millar Western, in Whitecourt, AB, Canada was performed.

Pulp is made from the cellulose fibres of wood chips. There are two basic ways to make pulp. The most common process reduces wood chips to their individual fibres through strong chemical treatment to produce a type of pulp called *kraft*. On the other hand, a combination of mild chemicals, heat and mechanical action is used to produce, Bleached Chemi-Thermo-Mechanical Pulp (BCTMP). The pulp produced in this way is also referred to as high-yield pulp, because the manufacturing process produces more pulp per tree than traditional pulping methods. Millar Western's pulp mill at Whitecourt produces pulp of this variety. The unit operations in this BCTMP process can be summarized as follows:

- Chipping: Softwood (Spruce, Pine and Fir) and Hardwood (Aspen) logs are converted into chips for ease of processing.
- Pre-treatment: The wood chips are screened and washed to remove debris. Screw-type presses then squeeze water and wood resins from the washed chips. Mild chemicals are added to soften the chips, which are then preheated to prepare them for the refining stage.
- Refining and Screening: After treatment and preheating, the chips pass through refiners where they are ground between large steel disks to separate their cellulose fibres, creating pulp. The wet, refined pulp is screened to sort the separated fibres from remaining fibre bundles.
- Cleaning and De-watering: The screened pulp passes through centrifugal cleaning cones. Heavier particles like sand and bark spin to the outside and are discharged. The lighter, clean pulp exits through the top of the cone, then passes through a disk filter to remove excess water before bleaching.
- Bleaching: The cleaned and filtered pulp is squeezed in presses and heated before entering bleach towers, where it is treated using hydrogen peroxide and caustic. The pulp is washed and pressed to extract bleach solution, which is recycled to the first stage of bleaching.
- Drying and Baling: The bleached pulp is fluffed to aid drying. In two stages, the fluffed pulp is flash-dried and blown through a series of cyclones before being pressed, compacted, wrapped, tied and loaded into railcars for delivery to customers.

The BCTMP mill at Whitecourt, consists of two parallel units which are nearly identical in design. These parallel units are named Line 1 and Line 2 respectively. They are subject to nearly the same stochastic environment. However, the production and operating conditions are different. Line 1 produces pulp which generally has lower brightness targets than Line 2 because it uses more of softwood as the raw material in contrast with Line 2 where the raw material is predominantly of the hardwood variety. This is illustrated in Fig. 5.1, where the plot has been re-scaled for confidentiality reasons. However, it is desired to have a single model for both the lines for reasons of long-term use. The reason for using a single model across two different units is that there might be operational constraints in the future which may require the use of either line to produce any of the currently produced grades and it is not possible from an economic viewpoint to re-identify models whenever there is a change in the operational strategy.

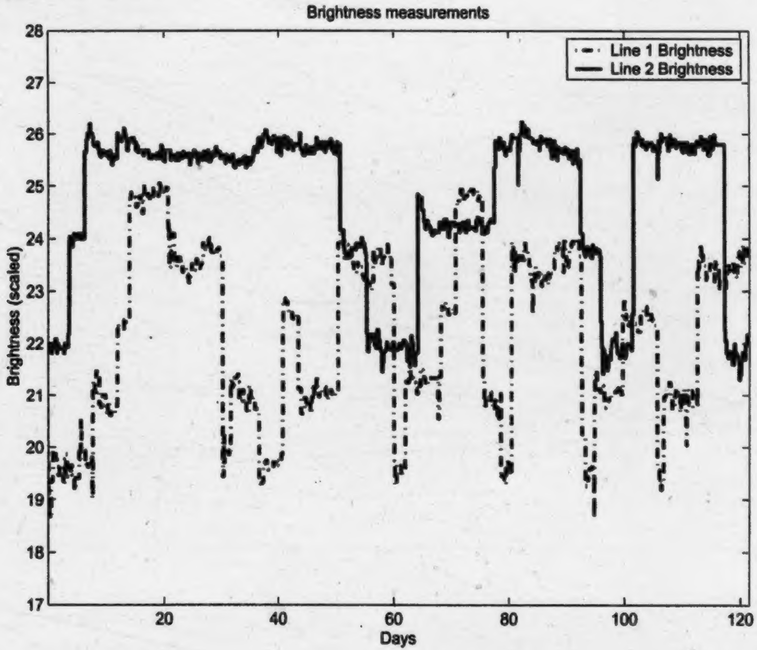


Figure 5.1: Line 1 and Line 2 Brightness measurements

A flow-sheet of the bleaching unit is shown in Fig. 5.2. A simplified version of this schematic is presented in Fig. 5.3. Each unit has four manipulated inputs, two measured disturbance variables and two outputs. The manipulated inputs are chemical add-rates (Peroxide and Caustic) to the two towers. There are two additional measurements (Aspen and Freeness) which are classified as measured disturbances. The outputs are the irregularly measured quality variables *viz.*, Brightness and Tensile strength of pulp. The forward path dynamics of the process can be captured by delay dominant low-order models. The unused chemicals from the towers are recycled back to the front-end of the bleach plant through the White-Water (WW) recycle stream. The presence of the recycle stream alters the dynamics of the process significantly. Hence it is necessary to take this into account while building a model for the process. Consequently, the average fraction of chemical being recycled must be captured in model. In addition there are significant nonlinear effects in the system. For instance, caustic addition affects the brightening effect of peroxide on the pulp though caustic is used predominantly for controlling the tensile strength of the pulp. Hence, there is some insight into the structure of the nonlinearity. In addition, it is not desirable to have completely black-box models (like those based on neural-networks) because of the difficulties in interpreting and explaining these models to the plant personnel.

Model structure: Based on the above description the following model structure of the system was chosen:

$$
\begin{aligned}
y_i(k) = {} & \sum_{q=1}^{4} \alpha_{qi} \sum_{p=1}^{m} a_r^{p-1} u_q(k - pT_{dq}) + \sum_{s=1}^{6} \sum_{q=1}^{4} \beta_{qsi} \sum_{p=1}^{m} a_r^{p-1} u_q(k - pT_{dq}) \\
& \times u_s(k - T_{ds} - (p-1)T_{dq}) + \sum_{s=1}^{6} \sum_{q=1}^{4} \gamma_{qsi} \sum_{p=1}^{m} a_r^{p-1} u_q(k - pT_{dq}) \qquad (5.1) \\
& \times (u_s(k - T_{ds} - (p-1)T_{dq}))^2 + \alpha_{5i} u_5 + \alpha_{6i} u_6 + v(k) \\
& i = 1, 2, \quad q \neq s, \quad a_r < 1
\end{aligned}
$$

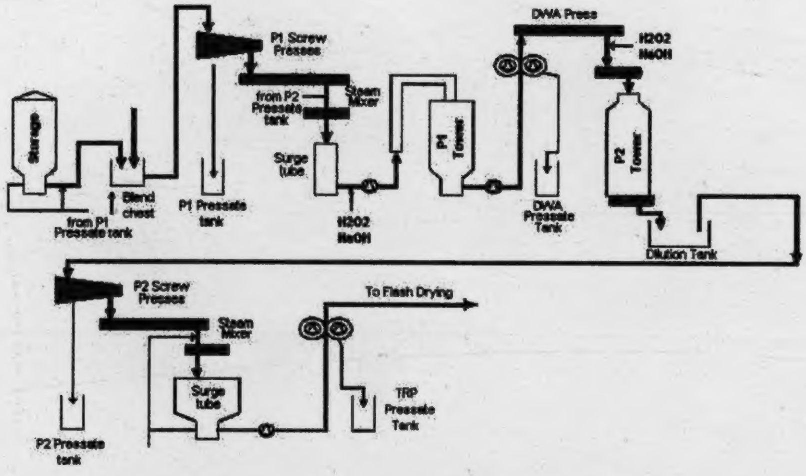

P1 Screw Presses
DWA Press
H2O2
NaOH
Storage
from P2 Pressate tank
Steam Mixer
Surge tube
P1 Tower
P2 Tower
Blend chest
from P1 Pressate tank
P1 Pressate tank
H2O2
NaOH
DWA Pressate Tank
Dilution Tank
P2 Screw Presses
To Flash Drying
Steam Mixer
Surge tube
TRP Pressate Tank
P2 Pressate tank

Figure 5.2: Flow sheet of BCTMP bleaching unit

In this representation, α_{qi} refers to the gain from the input terms, β_{qsi} refers to the gain from the products of pairs of inputs, γ_{qsi} refers to the gain from the products of inputs and squared input terms and m refers to the number of terms to which the FIR expansion can be carried out without any significant loss of information and a_r refers to the average fraction of chemical recycled. The formation of this model structure was possible after a number of rounds of data analysis and discussion with the process engineers.

The distribution of the sampling intervals, i.e., the time interval between consecutive samples, is shown for one of the output variables in Fig. 5.4. It is clear from the histogram

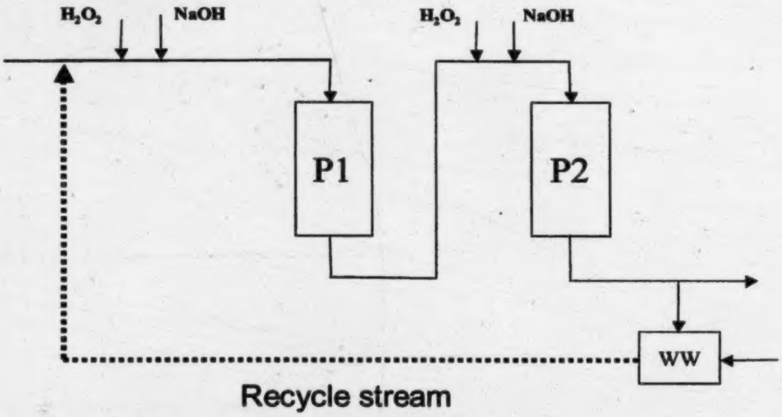


Figure 5.3: Simplified flow sheet of bleaching operation in a BCTMP mill

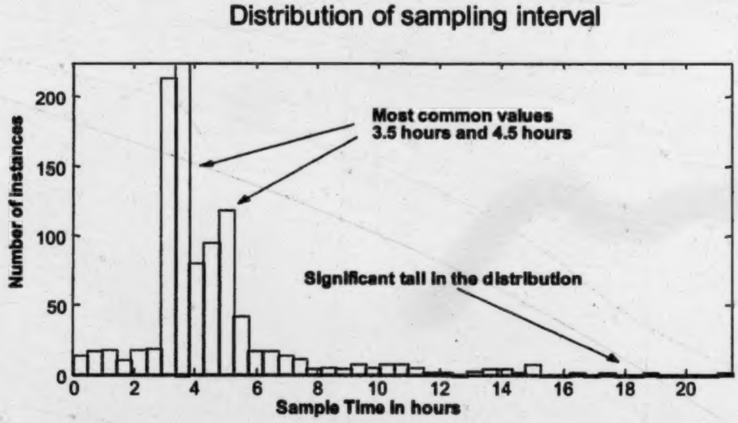


Figure 5.4: Distribution of the sampling intervals for the output variables

that the time interval between consecutive samples is a highly varying quantity. The most common values for this time interval are 3.5 or 4.5 hours. However, there are instances when this interval can take values as large as 20 hours. Given that the outputs are irregularly sampled, special interpolation strategies are required for generating the inter-sample output information. Alternatively, other work-around strategies, like constrained FIR-modeling or EM-algorithm based methods may be used. Throughout this exercise, routine operating data was used for model building. The main reason for using routine operating data was the prohibitive economic consequences of the loss of productivity caused by plant tests. It was feasible to use operating data in this exercise because it was found to contain enough excitation and a high signal-to-noise ratio (see Fig. 5.1). However it is important to note that routine operating data contains a number of correlations, including those indicative of the plant relationships and additional correlations introduced by control and operational strategies. The challenge is to identify useful models from process data. This is made easier if we have approximate knowledge of transport delays and gain directions.

From the process description, it is clear that the challenges in this identification problem include, using irregularly sampled data, accounting for delay dominated chemical recycle and nonlinear interactions. These are commonly encountered problems in chemical processes. In the following sections methods are described which can be used to overcome these problems.

5.4 Identification of processes with irregularly sampled outputs

This discussion is limited to the case where the input is sampled at a faster rate compared to the output because this is the commonly encountered situation in practice. The term "missing-data" is used to refer to the output value which is not available at a sampling instant when the input value is available.

Assume that the true process is of the form

$$\begin{aligned} \mathbf{x}_t &= \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1} + \mathbf{v}_t \\ \mathbf{y}_t &= \mathbf{C}\mathbf{x}_t + \mathbf{D}\mathbf{u}_t + \mathbf{w}_t \end{aligned} \tag{5.2}$$

where $\mathbf{A} \in \Re^{n \times n}, \mathbf{B} \in \Re^{n \times l}, \mathbf{C} \in \Re^{m \times n}$ and $\mathbf{D} \in \Re^{m \times l}$ are the system matrices and $\mathbf{x}_t \in \Re^n$ is the state vector. Assume that the system is stable. Additionally, assume that $\mathbf{u}(t) \in \Re^l$ and $\mathbf{y}(t) \in \Re^m$ are the input and output vectors respectively. $\mathbf{v}_t \in \Re^n$ and $\mathbf{w}_t \in \Re^m$ are uncorrelated white noise sequences *i.e.*,

$$E[\mathbf{w}_t\mathbf{w}_t^T] = \mathbf{Q}; \quad E[\mathbf{w}_t] = \mathbf{0}; \quad E[\mathbf{v}_t\mathbf{v}_t^T] = \mathbf{R}; \quad E[\mathbf{v}_t] = \mathbf{0}; \quad E[\mathbf{w}_t\mathbf{v}_t] = \mathbf{0} \quad \forall\, t \tag{5.3}$$

The time series data from $t = 1$ to $t = N$ for any variable is represented by $(.)_{1:N}$. The following notation is used for the expected values,

$$\mathbf{x}_t^s := E(\mathbf{x}_t|\mathbf{Y}_{1:s}); \quad \mathbf{P}_t^s := E(\mathbf{x}_t - \mathbf{x}_t^s)(\mathbf{x}_t - \mathbf{x}_t^s)^T; \quad \mathbf{P}_{t,t-1}^s := E(\mathbf{x}_t - \mathbf{x}_t^s)(\mathbf{x}_{t-1} - \mathbf{x}_{t-1}^s)^T \tag{5.4}$$

In addition, the following assumptions are made:

Assumptions

A1. Inputs are sampled uniformly every T units of time.

A2. Outputs are irregularly sampled and the maximum sampling times are $T_1, \cdots, T_n$ respectively.

A3. The input sampling time, is assumed to be the smallest sampling time *i.e.*, $T \leq T_i \quad \forall\, i$.

A4. The initial state is zero. This assumption is just for convenience. The results can easily be generalized for the case where the initial state is non-zero.

A5. The eigenvalues of A are strictly inside the unit circle. The pair $\{\mathbf{A}, \mathbf{C}\}$ is observable and $\{\mathbf{A}, \mathbf{B}\}$ is controllable.

A6. The system input is an arbitrary, quasi-stationary (Ljung 1999) deterministic sequence.

5.4.1 What is wrong with arbitrary interpolation?

The most common method of dealing with data which is irregularly sampled, is to interpolate between the known or observed data. This interpolation can be done in a number of ways, zero-order-hold interpolation, linear interpolation, quadratic interpolation etc. Similarly interpolation is used implicitly in some continuous-time identification methods. If there are not too many missing data points or, if the missing data points between any two observed data points are small in number then interpolations of above nature may not significantly affect the quality of the identified model. However, if the size of missing data points is significantly large compared to the number of observed data points, arbitrary interpolation can have an adverse effect on the identified model. In fact, there is no guarantee that the identified model will be consistent with the real process. Moreover, interpolation-based identification methods do not lead to statistically optimal models.

In Fig. 5.5, a typical plot of input-output data is shown. The two diamonds on the output signal indicate consecutive samples of the output. It is easy to notice that during that period of time, the input varies in a particular fashion. Now if this change in the input and the input changes before this period are ignored and arbitrary interpolation is performed to fill the missing data points, the filled missing data do not represent the real process. Hence, identification using arbitrary interpolation is like using *wrong* data and expecting to obtain the *right* model. Clearly, the optimal method of filling the missing data points is to use the *true* process model in estimating the missing data. This sounds

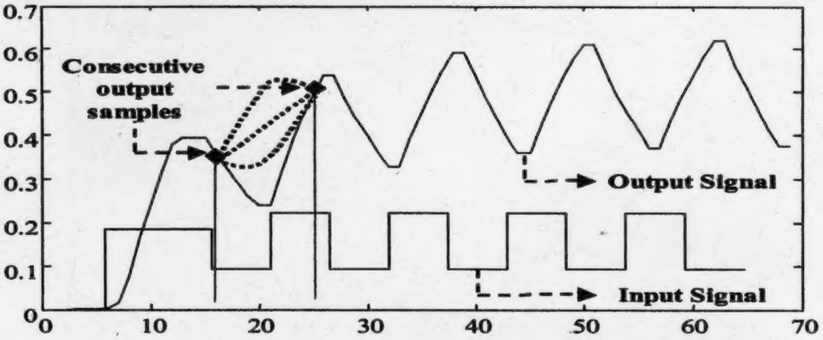


Figure 5.5: Various types of interpolation

illogical because identification of this *true* model is the problem under consideration. An intuitive alternative is to estimate a crude model and use it to estimate the missing data. Once the missing data is estimated, a complete data set can be created, which includes both the missing and observed data. It is then possible to use standard identification techniques on the complete data set. The new model obtained can then be used to re-estimate the missing data. This is the philosophy behind some of the approaches proposed in statistical literature to handle the missing-data identification problem.

5.4.2 Methods for handling missing data

Some of the possible approaches proposed to handle missing data while performing state and parameter estimation are discussed briefly. The most often used methods are (Little and Rubin 1987):

1. **List-wise Deletion**: A large amount of data is deleted in order to create single rate data subset. However, by deleting data we would lose valuable information about the process. Consequently the model quality is poor and this method can lead to the identification of biased models.

2. **Mean Imputation**: In this method a missing data value is replaced by the sample mean of the output calculated using a few samples around the missing sample or using all the available samples of the output. However, it is known to lead to biased estimates of sample quantities such as variances and covariances and hence may not be the appropriate solution.

3. **Maximum Likelihood Estimation (MLE)**: In this approach, the likelihood function of the observed data is maximized in order to estimate the model parameters. This is a very general method and hence maximum likelihood estimates can be developed for a variety of estimation problems. Once the likelihood function is set up it is usually solved using numerical optimization techniques like Newton-Raphson method etc. The advantage of performing MLE is that the estimates obtained have nice asymptotic properties like unbiasedness (when adjusted for degrees of freedom) and minimum variance. However, there are a few disadvantages associated with this approach. The computational effort involved may be too much and it is very easy to get stuck at local optima especially when there are too many missing samples. In addition though maximum likelihood estimates have good asymptotic properties, they can be heavily biased for small data sets and the estimates are sensitive to the initial guess. Hence methods such as bootstrapping (Shumway and Stoffer 2000) may be required to improve the estimates.

4. **Expectation Maximization (EM)**: The EM algorithm provides a simple and efficient approach to solve the MLE problem and is especially useful when there are missing samples. This is an iterative approach which involves an Expectation step (E-step) and a Maximization step (M-step). In each iteration, the E-step is used to obtain the expected value of the complete-data likelihood function conditioned on the available data and the estimated model parameters from the previous iteration. This is followed by the M-step in which a new set of model parameters which maximize the likelihood function obtained from the E-step is obtained. The disadvantages include the computational time and complexity. The E-step is implemented using Kalman filters, predictors and smoothers and hence may be computationally intensive.

5. **Multiple Imputation**: In multiple imputation (MI) a number of possible replacements are generated for each missing value, using Monte-Carlo simulations. Following this, parameter estimation is performed using standard techniques like MLE on each simulated complete dataset and the results are combined to produce estimates and confidence intervals that incorporate missing-data uncertainty. MI is similar to the EM algorithm and other computational methods for performing MLE based on the observed data alone. These methods use the likelihood function averaged over the distribution of the missing values. MI performs this same type of averaging by Monte Carlo rather than by analytical or numerical methods. In general, as the sample size increases, the inferences obtained by MI with sufficiently

> many imputations are nearly the same as those obtained by direct maximization of the likelihood. The reason for using Monte-Carlo simulations to generate the imputations is closely related to the probability model assumed for the complete data set. More often than not, these probability distributions tend to be complicated. Hence generating multiple imputations based on these models using analytical or numerical techniques (using methods like the EM algorithm) is complicated and often intractable. An attractive alternative is offered by the so-called *Markov chain Monte Carlo* (MCMC) methods that have appeared in statistical literature (Shumway and Stoffer 2000).

The above discussion concludes that MI and EM are optimal methods (in the MLE sense) to use for model identification from multi-rate or irregularly sampled data. When dealing with a linear-in-parameters model with a gaussian noise distribution, it is unnecessary to use the MI approach, unless one is interested in online recursive identification. If the computational load of the EM approach is considered too high, a constrained FIR model identification can be performed using gray-box identification tools to ensure that the model parameters obtained are reliable.

5.4.3 The EM algorithm

The EM algorithm is used to solve the problem of obtaining maximum likelihood estimates of model parameters. Details of the application of the EM algorithm for the estimation of the state-space system matrices are provided in the previous chapter. The expressions for the EM procedure for a model with the structure Eq. 5.2 are similar to those provided in Sec. 4.4.

5.4.4 How irregular can the sampling be?

One question which arises when using irregularly sampled data or multi-rate data for identification is whether there is enough information in the data set to identify the parameters of the model structure of interest. While the EM algorithm and other optimal methods provide a convenient way for handling missing data, they are still subject to sampling constraints. For identifying the parameters of a system, it is still necessary to ensure that no information has been lost due to the irregular sampling. A few simple models, which can be reconstructed without any loss of information from their discrete representations are considered with intuitive arguments to verify the sampling requirements.

1. **Steady-state model:** For identifying the parameters of a steady-state model, there are no restrictions on the sampling interval. The only restriction is in terms of the number of samples. As the number of samples increases, the properties of the estimates improve. Records containing missing data present two possibilities. One is to disregard these samples completely while identifying the parameters and then use these incomplete records and the identified model to obtain estimates of the missing data. The other possibility is to use the incomplete records to identify the relationships which are orthogonal to the relationships involving the missing data. This is automatically done if one uses an optimal procedure like the EM algorithm.

2. **Gain-plus-delay model:** There are two possibilities with this model structure. When the delay is known a priori, this model can be reduced to the steady-state model structure by time-shifting the input records. The procedure is more complicated in the case of multiple outputs. When the delay has to be estimated from the data, there could be estimation problems especially for multivariate systems and when one is dealing with closed-loop data. From a practical viewpoint, it is better to use transport delay estimates based on process knowledge rather than trying to find time delay estimates from data especially when dealing with routine operating data. This is because routine operating data contains a number of correlations, and one may be more interested in capturing the plant relationships rather than the other correlations.

3. **Gain-delay-recycle model:** Consider the system illustrated in Fig. 5.6. A simplified continuous-time representation of this system is shown in Eq. 5.5.

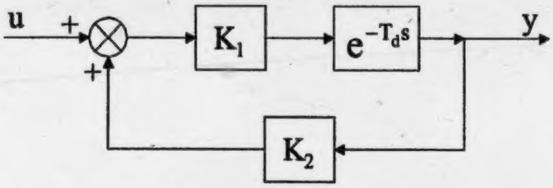


Figure 5.6: Block diagram of Gain-delay-recycle system

$$G(s) = \frac{K_1 e^{-T_d s}}{1 - K_1 K_2 e^{-T_d s}}, \quad K_2 < 1 \tag{5.5}$$

Conjecture 5.1 *Let $(T_s)_{max}$ be the largest allowed uniform sampling interval to avoid any loss of identifiability of the parameters K_1 and K_2. Let $(T_s)_{min}$ be the* *smallest sampling interval required to estimate the parameters K_1 and K_2. Then $(T_s)_{max} = (T_s)_{min} = T_d$.*

Remark 5.1 *To intuitively verify these arguments, consider the time domain representation of this system:*

$$y(t) = K_1 u(t - T_d) + K_2 y(t - T_d) \tag{5.6}$$

For this purely deterministic system, the parameters K_1 and K_2 can be estimated by considering two output samples, $y(t_1)$ and $y(t_2)$. Since the input is usually implemented using a known hold device, we can assume that the input value in Eq. 5.6 is available.

If the output sampling interval is $T_s = T_d$, then the parameters K_1 and K_2 can be estimated, for example, using the choice, $t_2 = t_1 + T_d$.

If the output is sampled at a fraction of T_d, i.e., $T_s = T_d/f$, f being a positive integer, it is still possible to solve for the parameters K_1 and K_2 uniquely.

Alternatively, let us assume that the output is sampled at a multiple of T_d, i.e., $T_s = f \times T_d$, f being a positive integer. For example, if the output is sampled at $T_s = 2T_d$, Eq. 5.6 can be written as:

$$y(t) = K_1 u(t - T_d) + K_1 K_2 u(t - 2T_d) + K_2^2 y(t - 2T_d) \tag{5.7}$$

$$= \gamma_1 u(t - T_d) + \gamma_2 u(t - 2T_d) + \gamma_3 y(t - 2T_d) \tag{5.8}$$

Unless a constraint is imposed on γ_2, it is possible to solve for γ_1, γ_2 and γ_3 by choosing $\gamma_2 = 0$. Hence it is necessary to impose a constraint on the coefficients if the sampling interval is chosen to be a multiple of T_d.

For the other cases, it is necessary to consider Eq. 5.6 as an infinite sum, i.e.,

$$y(t) = \sum_{r=0}^{\infty} K_1 K_2^r u(t - (r+1)T_d) \tag{5.9}$$

In this case, there will always be a finite truncation error when dealing with finite data lengths and hence it is not possible to estimate the parameters of the system perfectly even in the deterministic case unless additional restrictions are imposed on the input signals.

The above conjecture, that it is possible to recover the parameters of the system Eq. 5.6 when the sampling time is equal to the delay, is important in the current application. If this conjecture is true, the estimates of the parameters of the system obtained through constrained optimization techniques can be expected to be reasonably close to the true values given the fact that a majority of the sampling intervals in the system as shown in Fig. 5.4 are smaller than the delay in the system.

5.5 Identification of Delay-dominant recycle processes

Recycle systems consist of a forward path model and a recycle model with a positive feedback (Denn and Lavie 1982, Morud and Skogestad 1994). Hence, identification for recycle systems are similar to that of closed-loop systems. The dynamic behavior of recycle systems can be totally different from the that of systems with no recycle. Presence of recycle streams generally lead to a variety of interesting phenomena such as slow response, stair-case like step response and sensitivity to disturbances. Luyben (1994) has shown that recycle streams can sometimes lead to a *snowball effect* (a small change in the disturbance variable causes a large change in the manipulated variable) especially for certain control configurations. The behavior of recycle systems has been well studied in the literature and it is also known that due to their atypical behavior, special control algorithms are needed to achieve good closed loop performance. Chemical processes can exhibit large time delays. When identifying models for delay-dominant recycle processes the complexity of the dynamics should be taken into account. The use of black-box techniques may lead to models which fail to capture the characteristic staircase-shaped step-responses. In traditional identification methods, open-loop data is used for estimating unbiased models. Alternatively, there are methods for identifying models from closed-loop data (Forssell and Ljung 1999, Huang and Shah 1997). The problem in closed loop identification is to estimate both the process model and the controller using the set-point and process input/output data. This can be done in a number of ways *viz.,* two-step closed loop identification method, joint input-output identification and projection methods (Forssell and Ljung 1999). The problem of identification of recycle systems reduces to that of closed loop systems. However, recycle systems pose certain challenges in identification. Many traditional closed loop identification methods utilize three signals namely - set point, process input and process output. On the other hand, for recycle systems process input equivalent signal may not be available *i.e.,* there might be no measurements on the recycle stream. Hence, in general, it is not possible to obtain consistent estimates of the forward and recycle models if the input equivalent signal is not available.

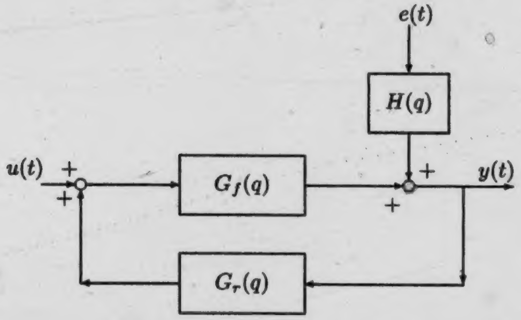


Figure 5.7: Block diagram of a typical recycle system

A typical recycle system is shown in Fig. 5.7 where $G_f(q)$ is the true forward process model and $G_r(q)$ is the true recycle model and $H(q)$ is the noise model. The process input is represented by $u(t)$, process output by $y(t)$ and the noise, represented by $e(t)$, is assumed to be normally distributed and white.

For the sake of simplicity the process is assumed to be SISO. q represents the forward shift operator. The estimated model consists of an estimated process model represented by $\hat{G}_f(q,\theta)$, an estimated recycle model represented by $\hat{G}_r(q,\theta)$ and an estimated noise model, $\hat{H}(q,\theta)$ where θ is the parameter vector. Assume that $\hat{H}$ and its inverse are both stable and that all the inputs and the outputs are quasi-stationary signals. Also assume that $\hat{G}$ and $\hat{H}$ are independent of the data length by defining the identified models as:

$$\begin{aligned}\hat{G} &= \lim_{N\to\infty} \hat{G}^N \\ \hat{H} &= \lim_{N\to\infty} \hat{H}^N\end{aligned} \tag{5.10}$$

where, $\hat{G}^N$ and $\hat{H}^N$ are models identified from data of length N.

Let the true model be (see Fig. 5.7)

$$\begin{aligned}y(t) &= G_f u(t) + G_f G_r y(t) + He(t) \\ &:= Pu(t) + Qy(t-d) + He(t) \\ &:= \Gamma U(k) + He(t)\end{aligned} \tag{5.11}$$

P and Q are defined in an obvious way. $d = d_f + d_r$ is the sum of the forward path delay, d_f and the recycle path delay, d_r and $U(t) = [u(t) \quad y(t-d)]^T$. Assume that $u(t)$ and $e(t)$

are independent. The following additional notation is introduced for convenience.

$$\tilde{\Gamma}(q) := \Gamma - \hat{\Gamma} \quad \& \quad \tilde{H}(q) := H - \hat{H} \tag{5.12}$$

$\tilde{P}$ and $\tilde{Q}$ are similarly defined. The data collected from identification experiments on the real process are denoted by $Z^N = \{u(1), y(1), \cdots, u(N), y(N)\}$. **Example:** A typical recycle system and its dynamic behavior is illustrated through this example. Consider the following recycle process:

$$G_f = \frac{0.9}{1 - 0.1q^{-1}} q^{-26} \quad G_r = \frac{0.25}{1 - 0.5q^{-1}} \tag{5.13}$$

Both forward and recycle models have relatively fast dynamics and the total delay in the loop is much larger than the settling time of the individual delay-free models. The step response of this process is shown in Fig. 5.8. The stair-case like step response is due to the large delay in the loop and the comparatively fast dynamics in the forward and recycle streams.

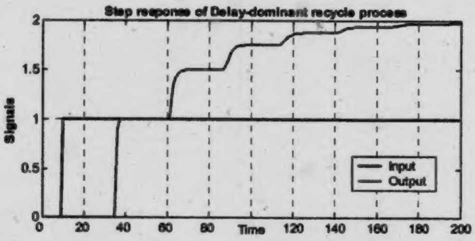


Figure 5.8: A typical staircase like step response of a recycle system

5.5.1 Estimation of consistent forward and recycle models

It is shown that by treating lagged output as one of the inputs, it is possible to obtain consistent estimates of the forward and recycle models for delay dominant processes. Consider the model equation in Eq. 5.11. Then the one-step ahead prediction error objective function for identification can be written as

$$\begin{aligned} V_N(\theta, Z^N) &= \frac{1}{N}\sum_{t=1}^{N}(y(t+1) - \hat{y}(t+1))^2 \\ &:= \frac{1}{N}\sum_{t=1}^{N}\varepsilon(t,\theta)^2 \end{aligned} \tag{5.14}$$

where, $\hat{y}(t+1)$ is the optimal one-step ahead predictor given by:

$$\hat{y}(t+1) = \hat{H}^{-1}\hat{\Gamma}U(t+1) + (1 - \hat{H}^{-1})y(t+1) \tag{5.15}$$

Then the prediction error estimate is given by:

$$\hat{\theta}_N = \underset{\theta \in D_{\mathcal{M}}}{\arg\min}\; V_N(\theta, Z^N) \tag{5.16}$$

Theorem 5.1

Consider the model in Eq. 5.11 and the estimation method in Eq. 5.16. Also assume that,

$$\tilde{P} := q^{-d_f} \sum_{k=0}^{n} p_k q^{-k} \qquad \tilde{H} := \sum_{k=0}^{m} h_k q^{-k}$$

where, $d \geq n + 1 \geq 1$ and $d \geq m \geq 1$. Then under the assumptions presented,

$$\hat{\theta}_N \rightarrow D_c = \underset{\theta \in D_{\mathcal{M}}}{\arg\min} \int_{-\pi}^{\pi} |\hat{H}|^{-2}|\tilde{P}|^2\Phi_u + |\hat{H}|^{-2}|\tilde{Q}|^2\Phi_{y_d} + |\hat{H}|^{-2}|\tilde{H}|^2\Phi_e d\omega \tag{5.17}$$

with probability one. Hence, it is possible to obtain consistent estimates of $\tilde{P}$ and $\tilde{Q}$.
Proof: As the data length, N, tends to infinity, the prediction error identification objective function in Eq. 5.14 converges almost surely to (Forssell and Ljung 1999),

$$\lim_{N\rightarrow\infty} V_N(\theta, Z^N) \rightarrow \bar{V}(\theta)$$

$$:= \frac{1}{2\pi}\int_{-\pi}^{\pi} tr\left(\begin{bmatrix} \tilde{\Gamma} & \tilde{H} \end{bmatrix} \begin{bmatrix} \Phi_U(\omega) & \Phi_{Ue}(\omega) \\ \Phi_{eU}(\omega) & \Phi_e(\omega) \end{bmatrix} \begin{bmatrix} \tilde{\Gamma} \\ \tilde{H} \end{bmatrix}^* (\hat{H} * \Phi_e * \hat{H}^*)^{-1}\right) d\omega$$

This objective function can be expanded to:

$$\bar{V}(\theta) := \frac{1}{2\pi}\int_{-\pi}^{\pi} tr\left(\begin{bmatrix} \tilde{P} & \tilde{Q} & \tilde{H} \end{bmatrix} \Omega \begin{bmatrix} \tilde{P} \\ \tilde{Q} \\ \tilde{H} \end{bmatrix}^* (\hat{H} * \Phi_e * \hat{H}^*)^{-1}\right) d\omega$$

where,

$$\Omega = \begin{bmatrix} \Phi_u(\omega) & \Phi_{uy_d}(\omega) & \Phi_{ue}(\omega) \\ \Phi_{y_du}(\omega) & \Phi_{y_d}(\omega) & \Phi_{y_de}(\omega) \\ \Phi_{eu}(\omega) & \Phi_{ey_d}(\omega) & \Phi_e(\omega) \end{bmatrix}$$

and y_d is the delayed output, $y(t-d)$. Notice that the minimum of this objective function occurs at the true parameters of the true models *i.e.*, when $\hat{P} = P, \hat{Q} = Q$ and $\hat{H} = H$. If Ω is non-diagonal, it is not possible to guarantee unbiased estimates of the process and noise models unless their correct structures are known. Typically, guessing the true structure of the noise model is very difficult and as a result, often, minimization of the above objective function leads to biased estimates. Let us take a closer look at the off-diagonal terms in the spectral matrix:

Term I: Assume that Φ_e is constant and that $\hat{H} = 1$ i.e., output-error structures are considered. Without loss of generality, assume that the input is white and hence the input spectrum is independent of frequency. Then one of the off-diagonal terms in the spectral matrix is

$$T_1 := \frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{H}^{-1}\tilde{P}\Phi_{uy_d}\tilde{Q}^*(\hat{H}^{-1})^*\Phi_e d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi} \tilde{P}\Phi_{uy_d}\tilde{Q}^*\Phi_e d\omega \tag{5.18}$$

Now note that,

$$\begin{aligned} y(t-d) &= \frac{P}{1-Qq^{-d}}q^{-d}u(t) + \frac{Hq^{-d}}{1-Qq^{-d}}e(t) \\ &:= q^{-d-d_f}\sum_{k=0}^{\infty} g_k q^{-k}u(t) + \frac{Hq^{-d}}{1-Qq^{-d}}e(t) \end{aligned} \tag{5.19}$$

for some impulse response coefficients, g_k. Therefore,

$$\begin{aligned} \Phi_{uy_d}(\omega) &= (\Phi_{y_d u}(\omega))^* \\ &= e^{i\omega(d+d_f)}\sum_{k=0}^{\infty} g_k e^{ik\omega}\Phi_u(\omega) \end{aligned} \tag{5.20}$$

Let us consider the case where $\tilde{P} = q^{-d_f}\sum_{k=0}^{n} p_k q^{-k}$ and define $\tilde{Q} = \sum_{k=0}^{\infty} q_k q^{-k}$ for some impulse response coefficients p_k and q_k. Then

$$\begin{aligned} T_1 &:= \frac{1}{2\pi}\int_{-\pi}^{\pi}\left(e^{-id_f\omega}\sum_{k=0}^{n} p_k e^{-ik\omega}\right)\left(e^{i(d+d_f)\omega}\sum_{k=0}^{\infty} g_k e^{ik\omega}\right)\left(\sum_{k=0}^{\infty} q_k e^{ik\omega}\right)\Phi_e\Phi_u d\omega \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\left(e^{id\omega}\sum_{k=0}^{n} p_k e^{-ik\omega}\right)\left(\sum_{k=0}^{\infty} g_k e^{ik\omega}\right)\left(\sum_{k=0}^{\infty} q_k e^{ik\omega}\right)\Phi_e\Phi_u d\omega \end{aligned} \tag{5.21}$$

Observe that if $d \geq n+1$, then the integrand in the above expression can be expressed as a summation of infinite exponentials with positive powers *i.e.,*

$$T_1 := \sum_{k=1}^{\infty}\frac{\Phi_e\Phi_u}{2\pi}\int_{-\pi}^{\pi} s_k e^{ik\omega}d\omega \tag{5.22}$$

for some constants s_k. The integral is zero for all positive k. Therefore, T_1 is zero if $d \geq (n+1)$ and

$$\tilde{P} = q^{-d_f}\sum_{k=0}^{n} p_k q^{-k} \tag{5.23}$$

Term II: Now consider the term involving $\Phi_{y_d e}$. This term can be written as

$$\begin{aligned} T_2 &:= \frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{H}^{-1}\tilde{Q}\Phi_{y_d e}\tilde{H}^*(\hat{H}^{-1})^*\Phi_e d\omega \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \tilde{Q}\Phi_{y_d e}\tilde{H}^*\Phi_e d\omega \end{aligned} \tag{5.24}$$

Let us now assume that

$$\tilde{H} := \sum_{k=0}^{m} h_k q^{-k}$$

i.e., the error in the impulse response coefficients of the noise model approaches zero after m impulse response coefficients. Then from Eq. 5.19,

$$\begin{aligned} \Phi_{y_d e} &= \frac{H(e^{i\omega})e^{-id\omega}}{1-Q(e^{i\omega})q^{-d}}\Phi_e(\omega) \\ &:= e^{-id\omega}\sum_{k=0}^{\infty} r_k e^{-ik\omega}\Phi_e(\omega) \end{aligned}$$

Hence,

$$\begin{aligned} T_2 &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\left(\sum_{k=0}^{\infty} q_k e^{-ik\omega}\right)\left(e^{-id\omega}\sum_{k=0}^{\infty} r_k e^{-ik\omega}\right)\left(\sum_{k=0}^{m} h_k e^{ik\omega}\right)\Phi_e^2(\omega)d\omega \\ &:= \sum_{k=1}^{\infty}\frac{\Phi_e\Phi_e}{2\pi}\int_{-\pi}^{\pi}\nu_k e^{-ik\omega}d\omega \end{aligned} \tag{5.25}$$

The above integral is again zero for all positive k. Due to the assumption that the input is not correlated with the noise, $\Phi_{eu}(\omega) = 0$ at all frequencies. Hence, the result in Eq. 5.17 follows. ■

Remark 1: The conditions on the forward model are clearly satisfied if it is a pure gain model with delay. They are approximately satisfied when the impulse response coefficients of $\tilde{P}$ decay very fast compared to the length of the total delay, d. It is straightforward to show that other cross terms in the spectral matrix are also zero under these conditions. If all the cross terms are zero, then the above objective function reduces to,

$$\bar{V}(\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} |\hat{H}|^{-2}|\tilde{P}|^2\Phi_u + |\hat{H}|^{-2}|\tilde{Q}|^2\Phi_{y_d} + |\hat{H}|^{-2}|\tilde{H}|^2\Phi_e d\omega \tag{5.26}$$

From this expression, it is clear that even if the noise model is fixed to unity, the minimum occurs at $\hat{P} = P$ and $\hat{Q} = Q$ provided the model set includes the true model.

It has been shown that for a delay-dominant recycle system with no measurements in the recycle stream, it is possible to obtain consistent estimates of the forward and recycle models by using the delayed output as a regressor. However, in the case of irregular sampling of the outputs, values of the lagged output might not be available where needed. In this case, identification of an FIR model is feasible. In order to ensure that the step response of the identified system conforms to the hypothesized pattern, it is necessary to perform constrained identification.

5.6 Other practical considerations

In this section other modifications required to improve the quality of the model are discussed.

5.6.1 Structure of the nonlinearity

It was found that there were significant nonlinear effects in the process, that needed to be accounted for in order to get good predictions. Following the recommendations of experts in the field of model identification (Ljung 1999) regarding physical interpretability of the model structure we restricted ourselves to simple structures as can be noted from Eq. 5.1. The physical reasoning for choosing such a model structure is that there is an interaction effect between sets of manipulated variables. For example it is known from practical experience that caustic addition affects the brightening ability of peroxide. Hence we chose the form:

$$\begin{aligned} y(t) = {} & a_0(u_1(t - T_{d1}) + a_r u_1(t - 2T_{d1}) + \cdots) \\ & + a_1(u_1(t - T_{d1}) u_2(t - T_{d2}) + a_r u_1(t - 2T_{d1}) u_2(t - T_{d1} - T_{d2}) + \cdots) \\ & + a_2(u_1(t - T_{d1}) u_2(t - T_{d2})^2 + a_r u_1(t - 2T_{d1}) u_2(t - T_{d1} - T_{d2})^2 + \cdots) \\ & + \cdots \end{aligned} \tag{5.27}$$

However, a number of tests using multiple data sets and physical reasoning were required before deciding whether the higher order terms were significant or not. Further, these model structures can be reduced to linear-in-parameters models and hence the identification of the parameters is similar to linear model identification.

5.6.2 Identification from routine operating data

There are many situations with the need for model identification without the luxury of plant tests. Under these circumstances, it might be necessary to get the best possible model using historical operating data. Even when it is necessary to perform plant tests, historical data can be used to build simple models for experimental design. This is especially important when the process has large settling times, like the BCTMP bleaching process. The use of historical data which contains enough excitation through grade changes etc., becomes significant. However, it is important to ensure that these models are physically meaningful. These models could easily reflect the moves made to the manipulated variables by the operator in response to changes in the output caused by unmeasured disturbances. Hence, the use of gray-box identification, physical reasoning and a critical evaluation of the model quality through infinite horizon predictions rather than *k-step ahead* predictions and long-term online testing becomes important. In the current application infinite horizon predictions were used for validation because the lab information system could not communicate with the main DCS system. However, the continued good performance of the infinite horizon predictions over more than 2 years enhances our confidence in the quality and reliability in the identified model.

5.7 Results

In this section, online brightness and tensile predictions from models developed using different techniques are presented. The performance of the identified models is evaluated using two indices, the cross-correlation (CC) and the root mean squared error (RMSE) which are defined as follows:

$$\mathrm{CC} = \frac{\mathrm{cov}(y, \hat{y})}{\sqrt{\mathrm{var}(y)\mathrm{var}(\hat{y})}}, \quad \mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} (y(i) - \hat{y(i)})^2} \tag{5.28}$$

5.7.1 Results using black-box identification

Initially black-box routines were used to develop models, without explicitly taking the recycle effect into account. These models did not separate the forward dynamics from the recycle dynamics. In addition, because black-box routines were used, an approximation of the overall dynamics of the process was obtained. For example, brightness predictions using a high order model obtained using *N4SID* a black-box subspace identification routine are shown in Fig. 5.9. In this plot, the first two-thirds of the samples were used for identification and the final one-third was the validation data set. The predictions look very good and hence one might be easily misled into believing that the models are good. In fact the converse is true. Even though the predictions are good, the model is not reliable because the step-responses, shown in Fig. 5.10, do not match with the expected fast-dynamics and

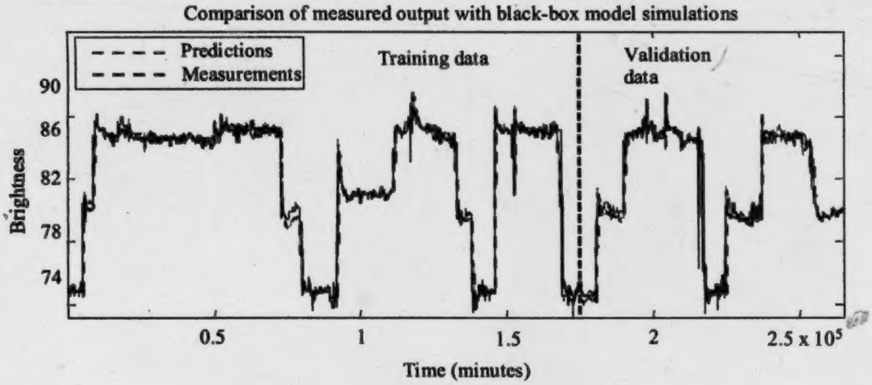


Figure 5.9: Predictions from N4SID look good

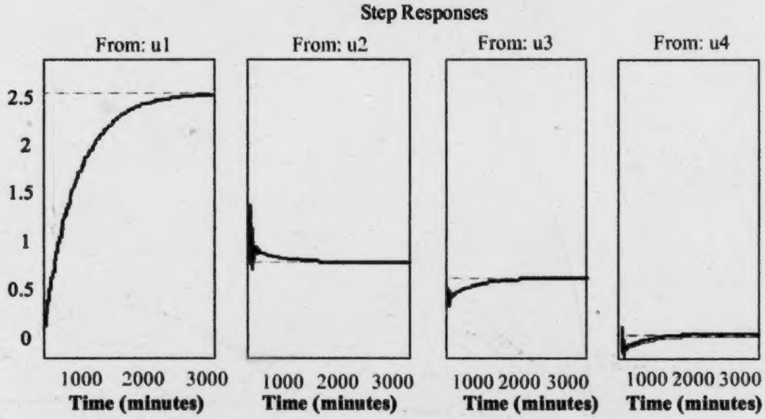


Figure 5.10: N4SID step responses are not satisfactory

characteristic staircase shape associated with the recycle effect.

5.7.2 Results using linear interpolation

Following this, it was decided to separate out the recycle effect by including a lagged output value as a regressor, as argued in Section 4. Hence $y(t - T_d)$ is used like an additional input. This leads to consistent models, provided that the impulse response of the delay-free forward model and the impulse response of the noise model die down in a time period less than the delay in the loop (Gopaluni *et al.* 2003). However, this method has the requirement that the output value $y(t - T_d)$ should be available. When the output is irregularly sampled,

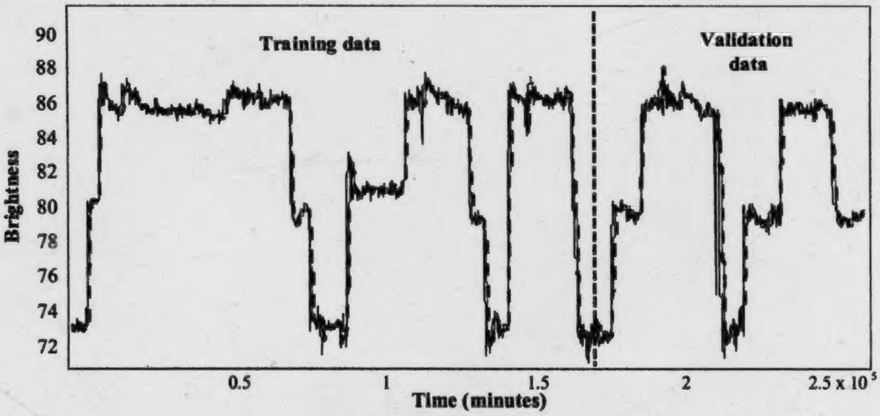


Figure 5.11: Predictions using linear interpolation look good

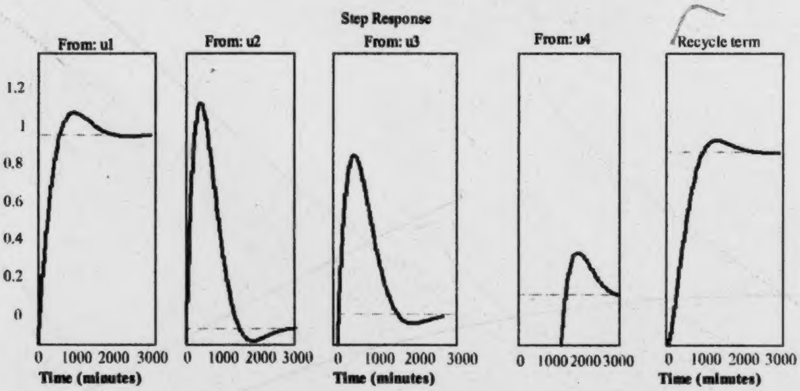


Figure 5.12: Step responses using interpolated data are not satisfactory

this requirement cannot be met always. A seemingly simple way to overcome this problem is to perform some kind of interpolation like linear interpolation to ensure the availability of the output value wherever it is required and not available. In retrospect, this leads to poor models, because the interpolation performed on the output is essentially univariate and does not take input variations into account. Brightness predictions using linear interpolation are shown in Fig. 5.11. Once again, the predictions look very good. However the models are far from satisfactory as is evident from the step-responses shown in Fig. 5.12. For example, the settling time for the forward paths have been identified to be of the order of 33 hours while in reality they are of the order of 20 minutes. Validation on a new data set (Fig. 5.13) shows a significant deterioration of predictions from the model based on linearly interpolated data.

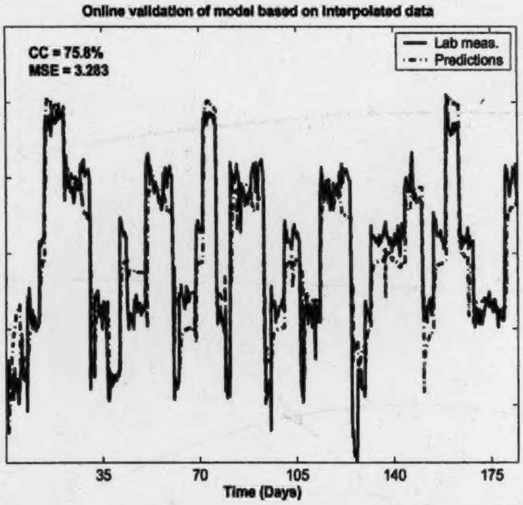


Figure 5.13: Deterioration of predictions for model based on linearly interpolated data

5.7.3 From black-box models to gray-box model model structures

In order to arrive at the final model structure, multiple iterations of analysis and discussions with the mill engineers were required. The observed dynamics were used to characterize this process as a time delay dominant recycle process. The following observations justify this hypothesis. The inputs are sampled every 10 minutes. The process consists of 2 towers (P1 and P2 tower). The chemical add-rates are manipulated upstream of each tower. It is observed that on changing the P1 add-rates, the output variables change 5 hours and 20 minutes later. On changing the P2 add-rates, the output variables change 3 hours and 40 minutes later. It is also observed that the direct change in the brightness and tensile is almost instantaneous, the change occurs within a couple of 10 minute sampling intervals, after the delays mentioned in the previous point. This characterizes the fast dynamics of the process. In addition to the direct change in the brightness and tensile there is a change because of the chemicals recycled to the front-end of the bleach plant. Given that the process can be characterized as a time-delay dominant recycle process, a strategy was needed to accommodate chemical recycling and irregular sampling. As shown previously, initial black-box models developed using Prediction Error Methods and Subspace Identification using interpolation techniques were not acceptable to the mill engineers because the dynamics portrayed by these models were unconvincing. Hence the model structure shown in Eq. 5.1 was chosen. Additionally, following the observation that by manipulating the caustic add-rate, the brightening effect of peroxide can be modified, nonlinearities were introduced into the model structure. It also prompted the investigation of other similar nonlinear interactions, which were validated on data sets over a long period of time.

5.7.4 Results using constrained FIR and EM algorithm

The final model structure chosen has the advantage of providing interpretable models as well as good predictions. This procedure involves the identification of constrained Finite Impulse Response (FIR) models as initial models. These models were later used as initial models in the EM algorithm following the procedure outlined in Ch. 4.

Structure of FIR models

FIR models of the form given in Eq. 5.29 were identified.

$$\begin{aligned} y(t) &= a_0 + a_1 u_1(t - T_{d1}) + a_1 a_R u_1(t - 2T_{d1}) + \cdots + a_1 a_R^{k-1} u_1(t - kT_{d1}) + \cdots \\ &+ a_2 u_2(t - T_{d2}) + a_2 a_R u_2(t - 2T_{d2}) + \cdots + a_2 a_R^{k-1} u_2(t - kT_{d2}) + \cdots \\ &+ \cdots + a_6 u_6(t - T_{d6}) + \nu(t) \\ &= G(q)u(t) + H(q)e(t) \end{aligned} \tag{5.29}$$

Here, a_0 represents an intercept term, a_m refers to the coefficient corresponding to the m^{th} input, a_R refers to the average chemical recycle fraction and T_{di} refers to the total delay in the forward and recycle paths from the i^{th} input to the output. k can be chosen depending on the number of past input values, which are hypothesized to affect the current value of the output.

Constrained identification of FIR models

The advantages of using an FIR structure for the initial model include the ability to handle irregularly sampled outputs, chemical recycling and input nonlinearity. The structure of the optimization problem to be solved is as follows:

$$\hat{\theta}_N = \arg\min_{\theta} \frac{1}{N} \sum_{t=1}^{N} \frac{1}{2} \epsilon^2(t, \theta) \quad \text{s.t. } f(\theta) \geq 0 \tag{5.30}$$

$f(\theta) \geq 0$ represents constraints applied on the identified parameters in order to obtain physically meaningful model parameters.

Following Eq. 5.29 and assuming an output-error structure for the model, the optimal one-step ahead prediction of y is given by:

$$\hat{y}(t|t-1) = H^{-1}(q)G(q)u(t) + [1 - H^{-1}(q)]y(t) = G(q)u(t) \tag{5.31}$$

This implies that the one-step ahead prediction is the same as the infinite horizon prediction. Similarly, the k-step ahead prediction of y is also equal to the infinite horizon prediction, $\hat{y}(t|t-k) = G(q)u(t)$. Hence the optimization problem simplifies to:

$$\hat{\theta}_N = \arg\min_{\theta} \frac{1}{N} \sum_{t=1}^{N} (y(t) - G(q,\theta)u(t))^2 \quad \text{s.t. } f(\theta) \geq 0 \tag{5.32}$$

$G(q)$ is defined according to Eq. 5.29. For example, it is necessary to impose the constraint $a_1 \geq 0$ because it is known that adding more peroxide, increases the brightness of the pulp. This constrained identification problem was solved using gray-box identification tools in Matlab©. Several trials with different initial guesses for the parameters were required to avoid convergence to local optima.

Robust estimation of average recycle fraction and selection of nonlinear effects

The term a_R in Eq. 5.29 represents the average recycle fraction. In order to obtain robust model estimates the problem of estimating a_R is decoupled from the problem of identifying the other parameters. Since, a_R is constrained to take a value between 0 and 1, it was varied between 0 to 1 in steps of 0.01, models identified for each of these values and the corresponding Root Mean Squared Error (RMSE) values were plotted. Following this, the model corresponding to the least RMSE was chosen. The estimated values of a_R and other model parameters depend on the set of chosen input factors and the interaction terms. For example, a list of the possible sets of direct factors and interaction terms for the brightness models is given below:

1. $\{\{u_1, \cdots, u_6\}$ or $\{u_1, u_2, (u_3 + u_4), u_5, u_6\}\}$.
2. $\{\{u_1, \cdots, u_6\}$ or $\{u_1, u_2, (u_3 + u_4), u_5, u_6\}\}$ and $\{u_1 \times u_5$ and $u_2 \times u_5\}$.
3. $\{u_1, \cdots, u_6, u_1 \times u_3, u_2 \times u_4\}$ or $\{u_1, u_2, (u_3 + u_4), u_5, u_6, (u_3 + u_4) \times u_1, (u_3 + u_4) \times u_2\}$.
4. $\{u_1, u_2, (u_3 + u_4), u_5, u_6, (u_3 + u_4)^2\}$.

In addition, combinations of the above cases can also be considered. In view of the variety of possible cases, the following procedure is used to obtain the best possible model:

- Choose a particular set of input factors.
- Find the model which gives the least RMSE value.
- Compare the RMSE values and find the best model structure.

A sample plot of the RMSE as a function of the average recycle fraction is shown in Fig. 5.14. In this case the chosen inputs were $u_1, u_2, (u_3 + u_4)$ and u_5.

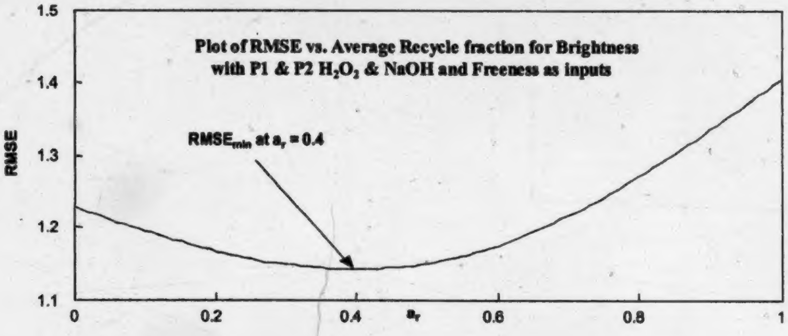


Figure 5.14: Choosing the best model using minimum RMSE

The FIR models identified for predicting Brightness and Tensile using this procedure are summarized below. For confidentiality reasons the coefficients have been re-scaled and the intercept term is masked.

$$\begin{aligned}
y_1(t) &= R(q)\{u_1(t-32) + 5.6u_2(t-22) + 6.4(u_3(t-32) + u_4(t-22)) \\
&\quad - 0.2u_1(t-32)(u_3(t-32) + u_4(t-22)) - 1.2u_2(t-22) \\
&\quad \times (u_3(t-32) + u_4(t-22))\} + 0.2u_5(t) + 0.01u_6(t) + \alpha_1 \\
y_2(t) &= R(q)\{-130u_1(t-32) + 30u_2(t-22) + 480u_3(t-32) + 430u_4(t-22) \qquad (5.33) \\
&\quad + u_1(t-32)u_6(t) - 0.4u_2(t-22)u_6(t) - 190u_1(t-32)u_3(t-32) \\
&\quad - 300u_2(t-22)u_4(t-22) + 76u_1(t-32)u_3(t-32)^2 \\
&\quad + 110u_2(t-22)u_4(t-22)^2\} - 18u_5(t) - 4u_6(t) + \alpha_2
\end{aligned}$$

where, $R(q) = (1 + a_R q^{-32} + a_R^2 q^{-64} + a_R^3 q^{-96} + a_R^4 q^{-128})$ and $a_R = 0.17$.

Given such a model, the staircase shaped response of an output can be simulated for a unit step change in an input. Such a response is shown for a step change in the peroxide add-rate in Fig. 5.15. It is easy to see from Eq. 5.33 that, due the interaction effects, the gain is a function of the absolute value of the inputs. For example, the brightening effect of

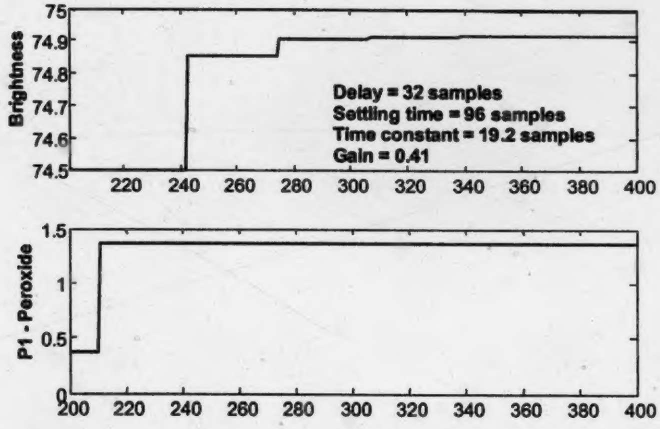


Figure 5.15: Step response of Brightness for unit change in P1 Peroxide

peroxide depends on the amount of caustic present in the tower. Hence the relation between u_1 and y_1 can be expressed as, $G_{11} = R(q)\,(1 - 0.2(u_3(t-32) + u_4(t-22)))\,u_1(t-32)$. Hence the gain from u_1 to y_1 is $K_{11} \approx (1.205) \times (1 - 0.2(u_3(t-32) + u_4(t-22)))$, which is a function of the caustic present in the tower. The FIR models identified using this gray-box procedure were used as initial models in the EM algorithm. In order to avoid an artificial increase in the system order while performing the state-space model identification (Amirthalingam *et al.* 2000), the inputs were time-shifted by the known delays (32 and 22 samples as is evident from Eq. 5.33).

Implementation

Predictors based on the models developed using the FIR-EM procedure have been implemented on the mills' distributed control system (DCS) and have been running satisfactorily for over an year. The implementation was done using Matrikon's ProcessACT© software, which provides an environment for design, development, simulation, and implementation of process control applications. A block diagram illustrating the implementation is shown in Fig. 5.16. Snapshots of the predictions using these models are shown in Fig. 5.17. These models are also being used in the development of an advanced process control scheme for controlling brightness and tensile to their desired targets.

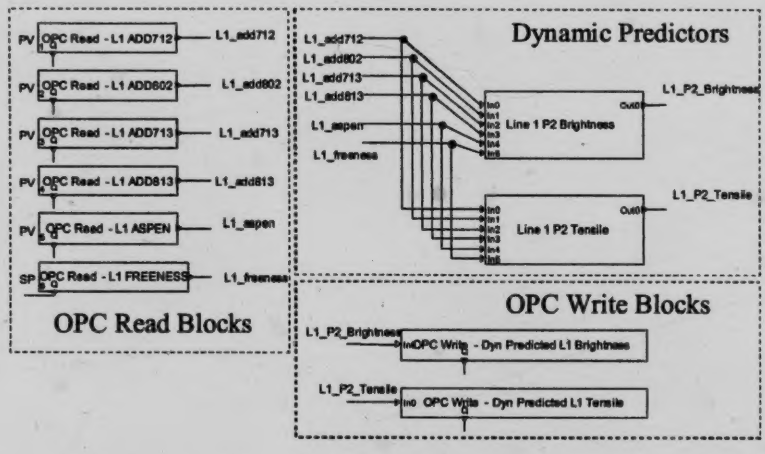

OPC Read Blocks
PV OPC Read - L1 ADD712
L1_add712
PV OPC Read - L1 ADD802
L1_add802
PV OPC Read - L1 ADD713
L1_add713
PV OPC Read - L1 ADD813
L1_add813
PV OPC Read - L1 ASPEN
L1_aspen
SP OPC Read - L1 FREENESS
L1_freeness
Dynamic Predictors
L1_add712
L1_add802
L1_add713
L1_add813
L1_aspen
L1_freeness
In0
In1
In2
In3
In4
In5
Line 1 P2 Brightness
Out0
L1_P2_Brightness
Line 1 P2 Tensile
L1_P2_Tensile
OPC Write Blocks
L1_P2_Brightness
In0 OPC Write - Dyn Predicted L1 Brightness
L1_P2_Tensile
In0 OPC Write - Dyn Predicted L1 Tensile

Figure 5.16: Block diagram showing implementation of predictors for Line 1

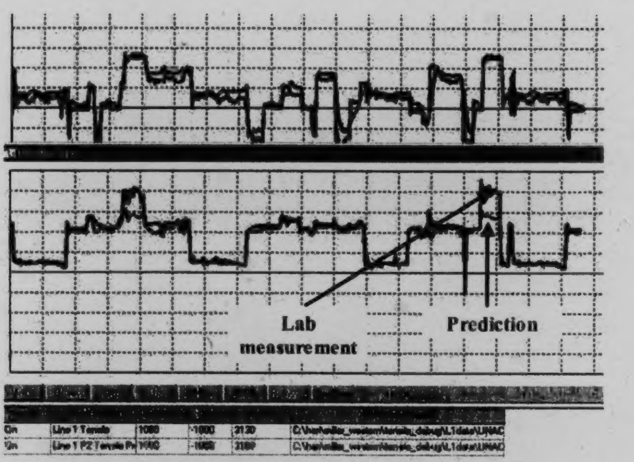
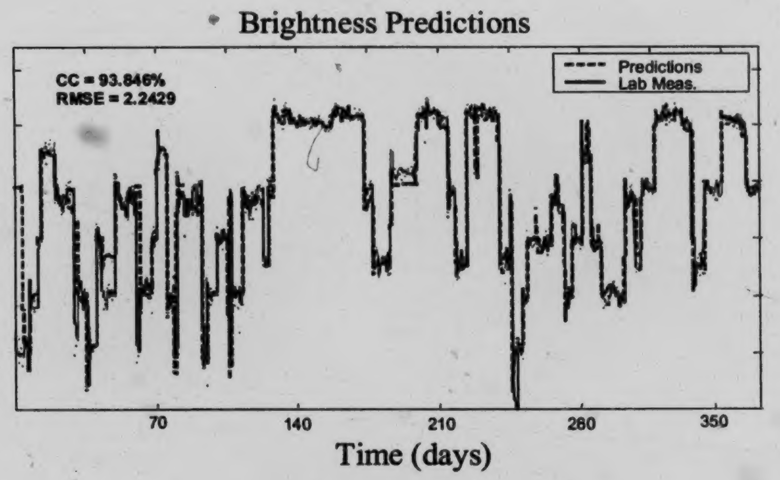

Lab measurement
Prediction

Figure 5.17: Snapshot of Line 1 predictions

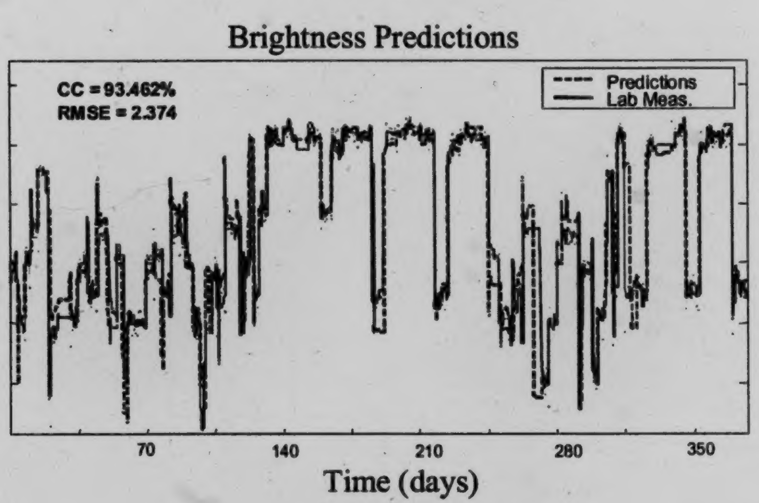
Online Predictions

The results of prediction of Line 1 and Line 2 Brightness and Line 1 and Line 2 Tensile using the constrained FIR-EM strategies are shown in Figs. 5.18, 5.19, 5.21 and 5.22 for a period of 12 months. The corresponding step responses are shown in Fig. 5.20. All these predictions are infinite horizon predictions. It is clear from the high values of the correlation coefficient and the low RMSE values (shown in the respective figures) that the predictions are quite satisfactory. In addition because of the imposition of the desired structure, the



Brightness Predictions

Figure 5.18: Line 1 Brightness Predictions



Brightness Predictions

Figure 5.19: Line 2 Brightness Predictions

model coefficients are interpretable and conform to what is known from prior knowledge. While it would have been insightful to include plots of the manipulated variables, we have not been able to do so because of confidentiality requirements.

Finally we present the tensile predictions for the period Jan - May 2003 (Figs. 5.23 and 5.24) to show that the models continue to have a satisfactory performance. These constitute

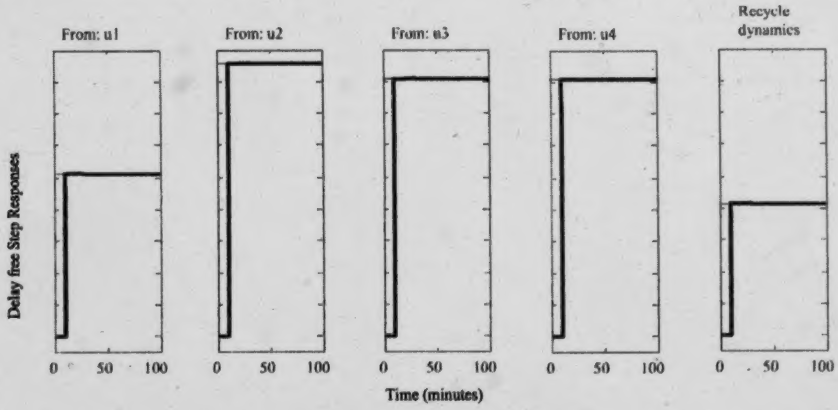


Figure 5.20: Step responses of FIR-EM Brightness model are satisfactory

a validation on a new data set and continue to be infinite horizon predictions.

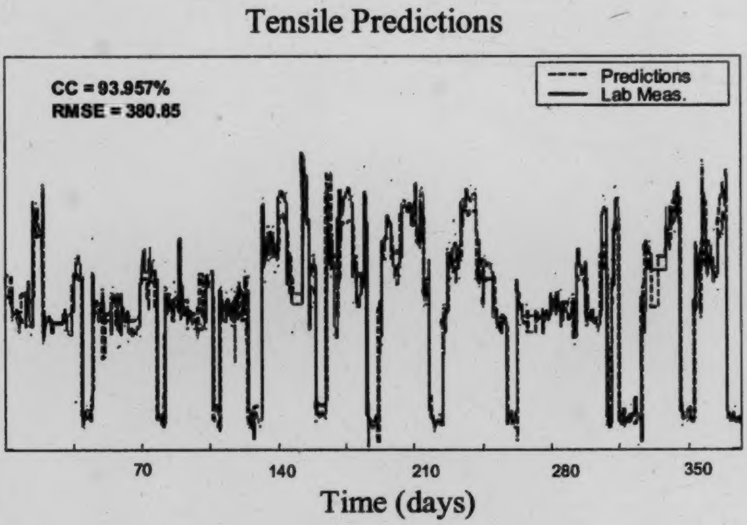


Figure 5.21: Line 1 Tensile Predictions

5.8 Conclusions

The application of gray-box techniques for model identification of the bleaching operation in a BCTMP process from routine operating data, has been presented. Novel techniques

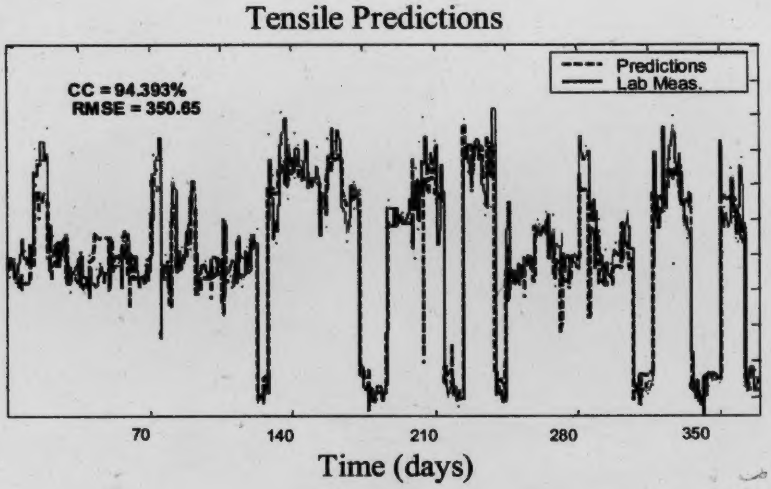


Figure 5.22: Line 2 Tensile Predictions

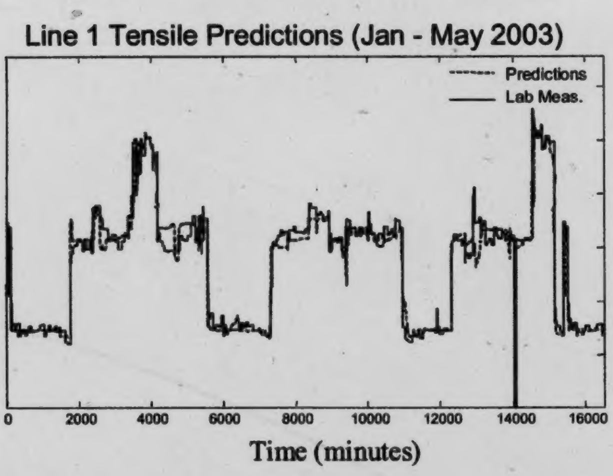


Figure 5.23: Line 1 Tensile Predictions (Jan - May 2003)

developed to deal with time-delay dominant recycle processes and irregularly sampled outputs have been described. The dangers of using arbitrary interpolation and black-box strategies have been demonstrated. Predictors based on the identified models have been used in the DCS for online prediction for over two years and are performing satisfactorily.

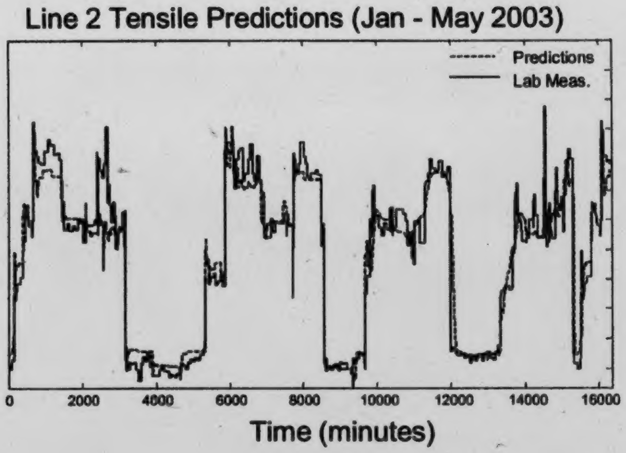


Figure 5.24: Line 2 Tensile Predictions (Jan - May 2003)

6

Diagnosis of additive process faults using continuous-time methods

6.1 Overview

In this chapter a novel subspace approach is proposed towards the identification of optimal residual models for process fault detection and isolation (FDI) in multivariate continuous-time systems. The problem is formulated in terms of the state space model of the continuous-time system. The motivation for formulating the problem in the continuous-time framework is that the fault gain matrix, which links the process faults to the system under consideration, is always available no matter how the faults vary with time. However, in the discrete-time domain, the fault gain matrix is only available when the faults are piece-wise constant within one sampling interval. To isolate faults, the fault gain matrix is essential[1].

The subspace algorithms are developed in the continuous-time domain to directly identify the residual models for FDI from sampled noisy data without separate identification of the system matrices. Furthermore, the proposed approach can also be extended in a straightforward manner towards the identification of the system matrices if they are needed. The newly proposed approach is applied to a simulated quadruple tank system, where leaks from any tanks are successfully detected and isolated. To make a comparison, the discrete-time residual models are also applied to the tank system for detection and isolation of leaks. It is demonstrated that the continuous-time FDI approach is practical and has better performance than the discrete-time FDI approach.

---

[1] Sections of this chapter have been presented or published as:

1. Li W., H. Raghavan and S.L. Shah. "Subspace identification of residual models for FDI in continuous-time systems". In Proc. of CHEMFAS-4, pages 224-229, Jejudo Island, Korea, 2001.
2. Li W., H. Raghavan and S.L. Shah. "Subspace identification of continuous time models for process fault detection and isolation". J. Proc. Cont., 13(5):407-421, 2003.

6.2 Introduction

In this chapter a novel subspace approach is proposed for the identification of optimal residual models for process fault detection and isolation (PFDI) in a multivariate continuous-time (CT) system. The problem is formulated in terms of the CT state space model, i.e. $\{A, B, C, D\}$ of the system, because most physical systems are CT by nature. Representation of CT systems by discrete-time (DT) models is only an approximation of their dynamics. Since it is easier to identify DT models than CT models and in many cases a CT system's dynamics can be represented well by a DT model, DT models have been widely used with success. However, in some cases, e.g. in the isolation of process faults, one has to use CT models. For example, while detection and isolation of sensor faults in a CT system can be performed using its discrete-time (DT) state space model (Chow and Willsky 1984, Li and Shah 2002, Qin and Li 2001), most of the done work on PFDI depends on the CT state space model (Ge and Fang 1988, Liu and Si 1997, Massoumnia 1986, White and Speyer 1987) of the system. Note that to perform PFDI, besides knowing the normal state space model: $\{A, B, C, D\}$ the fault gain matrix that links the process faults to the state variables of the system is indispensable. In this chapter it is shown that for the CT state space model of the system, the fault gain matrix is always available no matter how the faults are varying with time. However, in the DT state space model of the system, the fault gain matrix is available only when the faults follow some known functions of time, e.g. being piece-wise constant within one sampling interval. Without knowing the fault gain matrix, while process fault detection can still be carried out, process fault isolation is extremely difficult.

Most of the well known PFDI approaches, e.g. the ChowWillsky approach and the observer-based approaches (Frank 1990, Ge and Fang 1988, Liu and Si 1997, Massoumnia 1986, White and Speyer 1987, Willsky 1976) assume the availability of a CT state space model of the considered system. A frequency domain approach for fault detection in

DT systems is proposed in Zhang *et al.* (2003) where an analysis of the inter-sample behavior is performed in the frequency domain from the viewpoint of fault detection and isolation. In a limited number of cases, a CT state space model of the system can be obtained from first principles when the mechanisms of the system are well understood. However, in complex systems such as chemical engineering processes, building the CT state space models by means of the first principles is extremely difficult. Identifying an empirical process model by use of subspace methods is a good alternative. Subspace Identification has become an active research area. These methods have been successfully applied to identify multivariate DT state space models since the late 1980s (Moonen *et al.* 1989, Overschee and DeMoor 1996, Verhaegen and Dewilde 1992*a*, Verhaegen and Dewilde 1992*b*). In comparison with the traditional prediction errors methods (PEM) of identification (Ljung 1999), Subspace Identification Methods (SIM) have better numerical properties for systems with high dimensionality. For instance detailed remarks regarding the advantages of SIM over the PEM can be seen in (Viberg 1995). SIM has also been applied to the identification of primary residual models (PRM) for sensor fault detection and isolation (Li and Shah 2002, Qin and Li 2001). Recently, SIM for CT models has been proposed (Johansson *et al.* 1999). However, this approach has a drawback: it is sensitive to the initial values of CT signals. In this chapter, the numerical integrators proposed in (Sagara and Zhao 1990) are used to transform the signals and their derivatives in the CT domain into sampled data. An SIM is proposed and applied to the transformed signals for identification of the primary residual model (PRM) for process fault detection without identifying the system matrices: $\{A, B, C, D\}$, explicitly. The chosen numerical integrators are immune to initial values of the CT signals, computationally efficient, and have sufficient accuracy. Furthermore, the PRM is transformed into a set of structured residual models (SRMs) for fault isolation. The SRMs are designed such that each structured residual vector (SRV) generated by one SRM is most sensitive to only one single fault while remaining insensitive to the other faults.

The newly proposed approach is applied to a simulated tank system, where detection and isolation of leaks in any tanks has been successfully conducted. Moreover, although this approach is directed towards the identification of residual models, it can be easily extended to the identification of the complete system model $\{A, B, C, D\}$, if necessary. For example, in many control relevant problems, a knowledge of $\{A, B, C, D\}$ is desired.

The rest of this chapter is organized as follows. Sec. 6.3 is devoted to motivation and problem formulation. Sec. 6.4 outlines the numerical integrators to be used to transform the CT signals into DT data. Identification of the PRM for fault detection is given in

Sec. 6.5. In Sec. 6.6, identification of a set of SRMs for isolation of any single and multiple faults is investigated. The proposed approach is numerically evaluated in Sec. 6.7, where its effectiveness in detection and isolation of leaks in a simulated tank system is demonstrated. In addition, to make a comparison, the detection and isolation of leaks by means of the DT residual models has also been performed. This chapter ends in Sec. 6.8 with conclusions and remarks.

6.3 Motivation and Problem Formulation

6.3.1 Motivation

As depicted in Fig. 6.1, a four-tank system with leak $\delta_i(t)$ in the i^{th} tank, $\forall\ i \in [1,4]$, can be represented by the following equations (Johansson 2000):

$$\begin{aligned}
\frac{dh_1}{dt} &= \frac{\gamma_1 k_1}{A_1}v_1 - \frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_3}{A_1}\sqrt{2gh_3} - \frac{\epsilon_1\delta_1}{A_1}\\
\frac{dh_2}{dt} &= \frac{\gamma_2 k_2}{A_2}v_2 - \frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_4}{A_2}\sqrt{2gh_4} - \frac{\epsilon_2\delta_2}{A_2}\\
\frac{dh_3}{dt} &= \frac{(1-\gamma_2)k_2}{A_3}v_2 - \frac{a_3}{A_3}\sqrt{2gh_3} - \frac{\epsilon_3\delta_3}{A_3}\\
\frac{dh_4}{dt} &= \frac{(1-\gamma_1)k_1}{A_4}v_1 - \frac{a_4}{A_4}\sqrt{2gh_4} - \frac{\epsilon_4\delta_4}{A_4}
\end{aligned} \tag{6.1}$$

where, A_i - cross-section of tank i, a_i - cross-section of the i^{th} outlet hole, and h_i - water level in the i^{th} tank. Furthermore, in Eq. 6.1, v_i is the voltage applied to Pump i, and $k_i v_i$ is the corresponding flow. The parameters $\gamma_1, \gamma_2 \in (0,1)$ are determined from how the valves are set prior to an experiment. The flow to Tank 1 is $\gamma_1 k_1 v_1$, the flow to Tank 4 is $(1-\gamma_1)k_1 v_1$ and similarly for Tank 2 and Tank 3. The acceleration due to gravity is denoted by g. The measured level signals are $k_c h_1$, $k_c h_2$, $k_c h_3$ and $k_c h_4$, where k_c is a parameter associated with the sensor gain. In addition,

$$\epsilon_i = \begin{cases} 1, & \text{if the } i^{th} \text{ tank leaks.}\\ 0, & \text{if the } i^{th} \text{ tank does not leak.}\end{cases}$$

Eq. 6.1 can be linearized around an operating point (Johansson 2000): $\{h_1^\circ, h_2^\circ, h_3^\circ, h_4^\circ, v_1^\circ, v_2^\circ\}$ where h_i° and v_j°, $\forall$ $i \in [1,4]$ and $j \in [1,2]$, are the values of levels and voltages respectively, resulting in the following state-space model:

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \Xi_{d_i}\mathbf{f}_{d_i}(t)\\
\mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t)
\end{aligned} \tag{6.2}$$

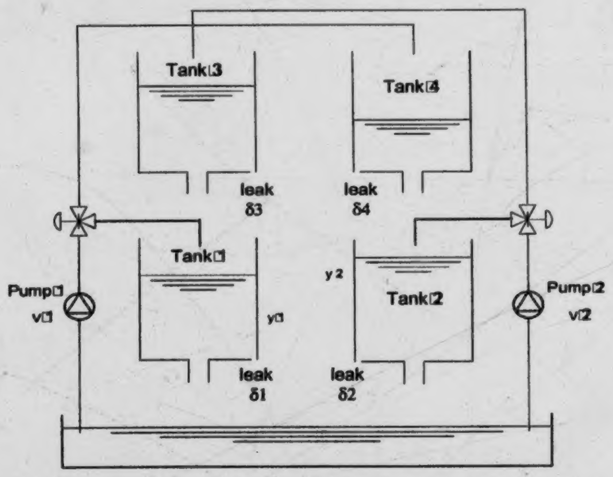


Figure 6.1: Schematic of the four water tank system

where, $\mathbf{x}(t) = [h_1^-(t)\ h_2^-(t)\ h_3^-(t)\ h_4^-(t)]^T$ is the state variable vector with $h_i^-(t) = h_i(t) - h_i^\circ$, $\forall\ i = 1, 4$, $\mathbf{u}(t) = [v_1(t) - v_1^\circ\ v_2(t) - v_2^\circ]^T$ is the input vector, $\mathbf{y}(t) = [k_c h_1^-(t)\ \ k_c h_2^-(t)\ k_c h_3^-(t)\ \ k_c h_4^-(t)]^T$ is the output vector and $\{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$ are system matrices, assuming that some or all the elements of $\mathbf{x}(t)$ are measurable. Details of $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ can be seen in (Johansson 2000). Further, in Eq. 6.2, $\mathbf{f}_{d_i}(t) \in \Re^{d_i}$ is the equivalent fault magnitude vector containing elements of $\left[\frac{-\delta_1^-(t)}{A_1} \cdots \frac{-\delta_4^-(t)}{A_4}\right]^T$ with Ξ_{d_i} standing for the associated columns of the 4×4 identity matrix $\mathbf{I}_4$, $\forall\ d_i \in [1, 4]$. Note that $\delta_i^-(t) = \delta_i(t) - \delta_i^\circ$ where δ_i° is the value of $\delta_i(t)$ at an operating point, $\forall\ i \in [1, 4]$.

Although Eq. 6.2 is derived for the four-tank system, it can be generalized to describe a wide class of multivariate linear time-invariant CT systems with process faults as follows:

$$
\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) + \Xi_{d_i}\mathbf{f}_{d_i}(t) \\
\mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)
\end{aligned}
\tag{6.3}
$$

where, $\mathbf{u}(t) \in \Re^q$, $\mathbf{y}(t) \in \Re^m$, $\mathbf{x}(t) \in \Re^n$, $\mathbf{f}_{d_i}(t) \in \Re^{d_i}$, $\Xi_{d_i} \in \Re^{n \times d_i}$, $\forall\ d_i \in [1, n]$, and $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ are similar to those defined in Eq. 6.2 (therein $\mathbf{D} = \mathbf{0}$) with compatible dimensions. It is assumed that $\{\mathbf{C}, \mathbf{A}\}$ is observable. Furthermore, in the sequel throughout this chapter, it is assumed that the order n is known, because schemes for the order determination in the subspace identification framework have been available, (Bauer 2001). Note that in Eq. 6.3, the fault model in CT domain is simply Ξ_{d_i}, i.e., columns of the identity matrix $\mathbf{I}_n \in \Re^{n \times n}$. However, as will be shown in the next paragraph, the fault model in the DT domain is not always available.

Discretizing Eq. 6.3 with a sampling period T_s leads to (Lewis 1986)

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}_d\mathbf{x}(k) + \mathbf{B}_d\mathbf{u}(k) + \mathbf{e}^{\mathbf{A}T_s}\int_0^{T_s} \mathbf{e}^{-\mathbf{A}\tau}\mathbf{\Xi}_{d_i}\mathbf{f}_{d_i}(kT_s+\tau)d\tau \\ \mathbf{y}(k) &= \mathbf{C}_d\mathbf{x}(k) + \mathbf{D}_d\mathbf{u}(k) \end{aligned} \tag{6.4}$$

where,

$$\begin{aligned} \mathbf{A}_d &= \mathbf{e}^{\mathbf{A}T_s}, \quad \mathbf{B}_d = \int_0^{T_s} \mathbf{e}^{\mathbf{A}\tau}\mathbf{B}d\tau \\ \mathbf{C}_d &= \mathbf{C}, \quad \mathbf{D}_d = \mathbf{D} \end{aligned} \tag{6.5}$$

In Eq. 6.4, under the assumption that $\mathbf{f}_{d_i}(kT_s+\tau)$ is constant within a sampling interval, i.e., $\mathbf{f}_{d_i}(kT_s+\tau) = \mathbf{f}_{d_i}(kT_s)$, $\forall\, \tau \in [0, T_s]$, the fault related term is equal to

$$\left(\mathbf{e}^{\mathbf{A}T_s}\int_0^{T_s} \mathbf{e}^{-\mathbf{A}\tau}\mathbf{\Xi}_{d_i}d\tau\right)\mathbf{f}_{d_i}(k)$$

where $\mathbf{f}_{d_i}(kT_s) = \mathbf{f}_{d_i}(k)$. Consequently, the fault model in the DT domain is

$$\mathbf{e}^{\mathbf{A}T_s}\int_0^{T_s} \mathbf{e}^{-\mathbf{A}\tau}\mathbf{\Xi}_{d_i}d\tau = \mathbf{A}_d\mathbf{A}^{-1}\left(\mathbf{I}_n - \mathbf{A}_d^{-1}\right)\mathbf{\Xi}_{d_i}$$

However, almost invariably this assumption will not hold, and therefore the fault-contributed term in the right hand side (RHS) of Eq. 6.4 is $\mathbf{e}^{\mathbf{A}T_s}\int_0^{T_s} \mathbf{e}^{-\mathbf{A}\tau}\mathbf{\Xi}_{d_i}\mathbf{f}_{d_i}(kT_s+\tau)d\tau$. In this case, since $\mathbf{f}_{d_i}(kT_s+\tau)$ is time varying, $\forall\, \tau \in [0, T_s]$, the fault model in the DT domain is not available. Without such a model, while fault detection can still be carried out, nothing can be done with respect to fault isolation. In particular, the assumption can be severely misleading if the sampling period T_s is relatively large. To obtain the DT fault model, one way is to increase the sampling frequency $\frac{1}{T_s}$ significantly. However, this can increase the cost of collecting data and cause numerical issues in system identification and control, as pointed out by Middleton and Goodwin (1990). This problem exists despite the tremendous improvement in data processing resources currently available. Therefore, for the purpose of process fault isolation in a CT system, a knowledge of its CT model instead of a DT model is essential. Note that even if the assumption of piece-wise constancy were to hold, as shown in the last equation above, a knowledge of $\mathbf{A}$ (the CT state matrix) is necessary for DT fault isolation.

6.3.2 Problem Formulation

Recursion of Eq. 6.3 yields

$$\mathbf{y}_s(t) = \mathbf{\Gamma}_s\mathbf{x}(t) + \mathbf{H}_s\mathbf{u}_s(t) + \mathbf{G}_s\mathbf{\Xi}_{s,d_i}\mathbf{f}_{s,d_i}(t) \tag{6.6}$$

where, $\Xi_{s,d_i} = \mathbf{I}_{s+1} \otimes \Xi_{d_i} \in \Re^{n_s \times d_i(s+1)}$ with $\otimes$ denoting the Kronecker tensor product,

$$\mathbf{y}_s(t) = \begin{bmatrix} \mathbf{y}(t) \\ \mathbf{y}^{(1)}(t) \\ \vdots \\ \mathbf{y}^{(s)}(t) \end{bmatrix} \in \Re^{m_s} \text{ is the stacked output vector,}$$

$$\mathbf{\Gamma}_s = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^s \end{bmatrix} \in \Re^{m_s \times n} \text{ is the extended observability matrix,}$$

and

$$\mathbf{H}_s = \begin{bmatrix} \mathbf{D} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & & \vdots \\ \vdots & \ddots & \ddots & \\ \mathbf{CA}^{s-1}\mathbf{B} & \ldots & \mathbf{CB} & \mathbf{D} \end{bmatrix} \in \Re^{m_s \times q_s}, \; \mathbf{G}_s = \begin{bmatrix} \mathbf{0} & \ldots & & \mathbf{0} \\ \mathbf{C} & \mathbf{0} & & \vdots \\ \vdots & \ddots & \ddots & \\ \mathbf{CA}^{s-1} & \ldots & \mathbf{C} & \mathbf{0} \end{bmatrix} \in \Re^{m_s \times n_s}$$

are two lower triangular block Toeplitz matrices. Note that $\mathbf{G}_s$ is completely dependent on the first ms rows of $\mathbf{\Gamma}_s$. Therefore, once $\mathbf{\Gamma}_s$ is identified, $\mathbf{G}_s$ can be derived from it. In addition, in Eq. 6.6, s is defined as the order of the parity space or the maximum detectability index of the fault (Liu and Si 1997) and is selected to be equal to n throughout this chapter without loss of generality; $\mathbf{y}^{(s)}(t)$ is the s^{th} derivative of $\mathbf{y}(t)$; $m_s = (s+1)m$; $n_s = (s+1)n$; and $q_s = (s+1)q$. The stacked vectors $\mathbf{u}_s(t) \in \Re^{q_s}$ and $\mathbf{f}_{s,d_i}(t)$ are in the similar formats to $\mathbf{y}_s(t)$. It is assumed that the order n is known.

The following definition is made for simplicity:

$$\tilde{\mathbf{H}}_s = \left[\; \mathbf{I}_{m_s} | \; -\mathbf{H}_s \; \right] \in \Re^{m_s \times (m_s + q_s)}$$

where $\mathbf{I}_{m_s} \in \Re^{m_s \times m_s}$ is an identity matrix. Hence Eq. 6.6 can be rewritten as

$$\tilde{\mathbf{H}}_s \begin{bmatrix} \mathbf{y}_s(t) \\ \mathbf{u}_s(t) \end{bmatrix} = \mathbf{\Gamma}_s \mathbf{x}(t) + \mathbf{G}_s \Xi_{s,d_i} \mathbf{f}_{s,d_i}(t) \tag{6.7}$$

Pre-multiplying Eq. 6.7 by a matrix $\mathbf{W}_0$, which lies in the null space of $\mathbf{\Gamma}_s$, i.e. $\mathbf{W}_0 \mathbf{\Gamma}_s = 0$, produces:

$$\boldsymbol{\varepsilon}_s(t) \equiv \mathbf{P}_s \begin{bmatrix} \mathbf{y}_s(t) \\ \mathbf{u}_s(t) \end{bmatrix} = \mathbf{W}_0 \mathbf{G}_s \Xi_{s,d_i} \mathbf{f}_{s,d_i}(t) \in \Re^{sn} \tag{6.8}$$

where $\mathbf{P}_s \equiv \mathbf{W}_0 \tilde{\mathbf{H}}_s = [\mathbf{W}_0 | - \mathbf{W}_0 \mathbf{H}_s]$ is defined as the PRM for fault detection. Note that $\mathbf{W}_0 \mathbf{G}_s$ is the fault model for the primary residual vector (PRV) $\boldsymbol{\varepsilon}_s(t)$.

By extending the Chow-Willsky scheme in the DT domain (Chow and Willsky 1984) to the CT domain, $\boldsymbol{\varepsilon}_s(t)$ is defined as the PRV in the CT domain, because it is zero if no fault occurs, i.e., $\mathbf{f}_{d_i}(t) = \mathbf{0}$; or nonzero if a fault occurs, i.e., $\mathbf{f}_{d_i}(t) \neq \mathbf{0}$. In order to ensure that the process fault $\mathbf{f}_{d_i}(t)$ is detected with highest sensitivity, $\mathbf{W}_0$ is designed to have maximum covariance with $\mathbf{G}_s$ while being orthogonal to $\boldsymbol{\Gamma}_s$. Mathematically the following objective functions are established, $\forall\, i,j = \{1, \mathrm{Rank}(\mathbf{G}_s)\} \cap \{i \neq j\}$.

$$J_0^i = \max_{\mathbf{W}_0(i,:)} \mathbf{W}_0(i,:)\mathbf{G}_s\mathbf{G}_s^T\mathbf{W}_0^T(i,:) - \mathbf{W}_0(i,:)\boldsymbol{\Gamma}_s\boldsymbol{\lambda}_0^i - \lambda_1^i(\|\mathbf{W}_0(i,:)\|_2 - 1) - \lambda_2^i\mathbf{W}_0(i,:)\mathbf{W}_0^T(j,:) \tag{6.9}$$

where $\mathbf{W}_0(i,:)$ and $\mathbf{W}_0(j,:)$ are the i^{th} and j^{th} rows of $\mathbf{W}_0$, respectively; and $\boldsymbol{\lambda}_0^i \in \Re^n$, λ_1^i and λ_2^i, are the Lagrangian multipliers; and $\| \ \|_2$ is the 2-norm of a vector. Note that in the objective functions, the conditions $\mathbf{W}_0(i,:)\boldsymbol{\Gamma}_s = \mathbf{0}$ and the orthonormality among the rows of $\mathbf{W}_0$ have been taken into consideration.
Similar to (Li and Shah 2002), the optimal solution to $\mathbf{W}_0$ can be obtained as follows:

$\mathbf{W}_0^T$ = *Eigenvectors associated with non-zero eigenvalues of the matrix* $\boldsymbol{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T$

where $\boldsymbol{\Gamma}_s^{\perp} = \mathbf{I}_{m_s} - \boldsymbol{\Gamma}_s\boldsymbol{\Gamma}_s^{+}$ and $^+$ indicates the Moore-Penrose pseudo inverse of the argument. Note that $\boldsymbol{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T$ is a non-symmetric matrix. To calculate $\mathbf{W}_0$, this non-symmetric eigenproblem is first translated into a symmetric eigenproblem. Then existing algorithms can be used. The calculation of $\mathbf{W}_0$ is given in the Appendix. Since $\boldsymbol{\Gamma}_s$ has rank n, $\boldsymbol{\Gamma}_s^{\perp}$ will have rank $m_s - n$. Furthermore, note that $\mathbf{G}_s\mathbf{G}_s^T$ has a rank equal to $\mathrm{Rank}(\mathbf{G}_s)$. As a result, $\mathbf{W}_0$ will have $min\{m_s - n, \mathrm{Rank}(\mathbf{G}_s)\}$ non-zero eigenvalues. Assume that $m = n$ and $\mathrm{Rank}(\mathbf{C}) = n$. This leads us to conclude that $\mathrm{Rank}(\mathbf{G}_s) = sn$ and $min\{m_s - n, \mathrm{Rank}(\mathbf{G}_s)\} = sn$, which is the number of independent rows in $\mathbf{W}_0$, i.e., $\mathbf{W}_0 \in \Re^{sn \times m_s}$.

Assume that both the sampled inputs $\mathbf{u}^{\circ}(k)$ and outputs $\mathbf{y}^{\circ}(k)$ are corrupted by independently identically distributed white noise vectors with gaussian densities $\mathbf{v}(k) \in \Re^q$ and $\mathbf{o}(k) \in \Re^m$, which have covariance matrices $\mathbf{R}_v$ and $\mathbf{R}_o$ respectively, i.e.,

$$\mathbf{u}^{\circ}(k) = \mathbf{u}(k) + \mathbf{v}(k), \ \ \mathbf{y}^{\circ}(k) = \mathbf{y}(k) + \mathbf{o}(k) \tag{6.10}$$

This is referred to as the EIV case (Chou and Verhaegen 1997). Note that the two noise vectors are mutually independent and are independent of $\mathbf{u}(k)$ and $\mathbf{y}(k)$. The problem of residual model identification can then be stated as follows:

Given the sampled inputs and outputs when $\mathbf{f}_{d_i}(t) = \mathbf{0}$, *identify the PRM:* $\mathbf{P}_s$, *which includes the consistent identification of* $\mathbf{\Gamma}_s$, *calculation of* $\mathbf{W}_0$, *and identification of* $\mathbf{W}_0\mathbf{H}_s$. *Furthermore, design a set of SRMs for fault isolation from the PRM.*

6.4 Overview of the Numerical Integrators

To identify $\mathbf{\Gamma}_s$ and $\mathbf{W}_0\mathbf{H}_s$, $\mathbf{u}_s(t)$ and $\mathbf{y}_s(t)$, which contain the CT signals $\{\mathbf{u}(t), \mathbf{y}(t)\}$ and their derivatives from the 1^{st} up to the s^{th} order are needed. The derivatives are not directly measurable and many approaches have been developed to deal with them (Whitfield and Messali 1987, Johansson 1994, Chou *et al.* 1999). In this work, the numerical integrator proposed in (Sagara and Zhao 1990) is used to transform the derivatives into discrete data. This integrator has a number of attractive features: simplicity of implementation, insensitivity to initial values of the CT signals, and high accuracy.

The integral of a CT signal, e.g., $\mathbf{u}(t)$, over the time interval $[t - lT_s, t)$ can be approximately calculated by

$$\zeta_1(\mathbf{u}(t)) = \int_{t-lT_s}^{t} \mathbf{u}(\tau)d\tau \approx \omega_0\mathbf{u}(t) + \ldots + \omega_l\mathbf{u}(t - lT_s) = \sum_{i=0}^{l} \omega_i q^{-i}\mathbf{u}(t) \tag{6.11}$$

where T_s is the integration step size, chosen to be the same as the sampling interval for ease of implementation; l is considered as the length factor of the integrator (a natural number); and q^{-1} is the unit delay operator, i.e., $q^{-1}\mathbf{u}(t) = \mathbf{u}(t - T_s)$. The filter coefficients ω_i depends on the type of numerical integration methods to be employed. For instance, when the trapezoidal integration rule is used, the filter coefficients are (Sagara and Zhao 1990):

$$\omega_0 = \omega_l = \frac{T_s}{2}, \quad \omega_i = T_s, \quad i = 1 \cdots l - 1$$

Similarly, the s^{th} multiple integral of the i^{th} derivative $\mathbf{u}^{(i)}(t)$ of $\mathbf{u}(t)$ can be defined as follows:

$$\zeta_s\left(\mathbf{u}^{(i)}(t)\right) = \int_{t-lT_s}^{t} \int_{\tau_1-lT_s}^{\tau_1} \cdots \int_{\tau_{s-1}-lT_s}^{\tau_{s-1}} \mathbf{u}^{(i)}(\tau)d\tau_s \ldots d\tau_1, \quad i = 0 \cdots s. \tag{6.12}$$

Furthermore, Eq. 6.12 can be approximately calculated by

$$\zeta_s\left(\mathbf{u}^{(i)}(t)\right) = \phi_{s,i}\left(q^{-1}\right)\mathbf{u}(t), \quad i = 0 \cdots s. \tag{6.13}$$

where,

$$\phi_{s,i}\left(q^{-1}\right) = \left(1 - q^{-l}\right)^i \left(\omega_0 + \omega_1 q^{-1} + \cdots + \omega_l q^{-l}\right)^{s-i} = \sum_{\nu=0}^{sl} \beta_\nu^i q^{-\nu}$$

with β_ν^i being the ν^{th} coefficient in the polynomial $\phi_{s,i}(q^{-1})$. A rigorous proof of Eq. 6.13 and comments on the numerical integrator can be found in Sagara and Zhao (1990). Note that the optimal choice of the filter length is an open issue especially for multivariate systems. The approach of minimizing the prediction error (Wang and Gawthrop 2001), might provide a solution.

6.5 Identification of the PRM

This section is mainly devoted to the identification of the PRM, which can be directly obtained by performing a QR decomposition followed by a singular value decomposition (SVD) on an augmented data matrix. In addition, identification of the system matrices $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ is investigated, because in many situations a knowledge of these matrices is necessary. As will be shown later, identification of $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ depends on $\mathbf{\Gamma}_s$ and $\mathbf{H}_s$. The key to the identification of the PRM is to obtain consistent estimates of $\mathbf{\Gamma}_s$ and $\mathbf{H}_s$ first.

6.5.1 Description of the Identification problem

When $\mathbf{f}_{s,d_i}(t) = 0$, Eq. 6.6 is reduced to

$$\mathbf{y}_s(t) = \mathbf{\Gamma}_s\mathbf{x}(t) + \mathbf{H}_s\mathbf{u}_s(t) \tag{6.14}$$

Performing multiple integration of Eq. 6.14 s times by using Eq. 6.13 yields,

$$\phi_s\left(q^{-1}\right)\mathbf{y}(t) = \mathbf{\Gamma}_s\phi_s\left(q^{-1}\right)\mathbf{x}(t) + \mathbf{H}_s\phi_s\left(q^{-1}\right)\mathbf{u}(t) + \mathbf{e}_s(t, T_s) \tag{6.15}$$

where,

$$\phi_s\left(q^{-1}\right)\mathbf{y}(t) = \begin{bmatrix} \phi_{s,0}\left(q^{-1}\right) \\ \phi_{s,1}\left(q^{-1}\right) \\ \vdots \\ \phi_{s,s}\left(q^{-1}\right) \end{bmatrix} \otimes \mathbf{y}(t) \in \Re^{m_s}, \quad \phi_s\left(q^{-1}\right)\mathbf{u}(t) = \begin{bmatrix} \phi_{s,0}\left(q^{-1}\right) \\ \phi_{s,1}\left(q^{-1}\right) \\ \vdots \\ \phi_{s,s}\left(q^{-1}\right) \end{bmatrix} \otimes \mathbf{u}(t) \in \Re^{q_s}$$

and $\mathbf{e}_s(t, T_s)$ is a truncation-error due to numerical integration of inputs and outputs. With available sampled data $\{\mathbf{u}^\circ(k), \mathbf{y}^\circ(k)\}$, substituting Eq. 6.10 into Eq. 6.15 with $t = kT_s$ yields

$$\phi_s\left(q^{-1}\right)\mathbf{y}^\circ(k) = \mathbf{\Gamma}_s\phi_s\left(q^{-1}\right)\mathbf{x}(k) + \mathbf{H}_s\phi_s\left(q^{-1}\right)\mathbf{u}^\circ(k) + \mathbf{E}_s(k) \tag{6.16}$$

where

$$\mathbf{E}_s(k) = -\mathbf{H}_s\phi_s\left(q^{-1}\right)\mathbf{v}(k) + \phi_s\left(q^{-1}\right)\mathbf{o}(k) + \mathbf{e}_s(k, T_s) \in \Re^{m_s}$$

The first term in $\mathbf{E}_s(k)$ is a moving average (MA) process of $\mathbf{v}(k)$ and $\mathbf{o}(k)$. In addition, as addressed in (Sagara and Zhao 1990), the truncation error $\mathbf{e}_s(k, T_s)$ can be controlled with the sampling interval T_s. For simplicity, in the following analysis, it is assumed that a small T_s is used so that the influence of $\mathbf{e}_s(k, T_s)$ on the identification of $\mathbf{P}_s$ can be negligible, i.e., $\mathbf{e}_s(k, T_s) \approx \mathbf{0}$.

The following data matrix is formed from the inputs $\mathbf{u}^{\circ}(k)$

$$\mathbf{U}^{\circ}_{k,s,N} = \left[\begin{array}{ccc} \phi_s\left(q^{-1}\right)\mathbf{u}^{\circ}(k) & \cdots & \phi_s\left(q^{-1}\right)\mathbf{u}^{\circ}(k+N-1) \end{array}\right] \in \Re^{q_s \times N}$$

where N is the number of data samples in the matrix. Similarly, the output data matrix $\mathbf{Y}^{\circ}_{k,s,N} \in \Re^{m_s \times N}$ is formed. The use of the newly formed data matrices in Eq. 6.16 gives

$$\mathbf{Y}^{\circ}_{k,s,N} = \mathbf{\Gamma}_s \mathbf{X}_{k,N} + \mathbf{H}_s \mathbf{U}^{\circ}_{k,s,N} + \mathbf{E}_{k,s,N} \in \Re^{m_s \times N} \tag{6.17}$$

where,

$$\mathbf{X}_{k,N} = \phi_s\left(q^{-1}\right)\left[\mathbf{x}(k) \cdots \mathbf{x}(k+N-1)\right] \in \Re^{n \times N}$$

and $\mathbf{E}_{k,s,N} \in \Re^{m_s \times N}$ resembles $\mathbf{U}^{\circ}_{k,s,N}$ in format.

After the establishment of Eq. 6.17, the following remarks can be made:

Remark 6.1 The data matrices $\mathbf{U}^{\circ}_{k,s,N}$ and $\mathbf{Y}^{\circ}_{k,s,N}$ are composed of sampled inputs and outputs and their time-lagged values, respectively. Further, each data matrix has $s+1$ row blocks and N columns. For example, in the i^{th} column of $\mathbf{U}^{\circ}_{k,s,N}$, one row block contains the linear combination of $\mathbf{u}^{\circ}(k+i-1)$ and the time-lagged values $\{\mathbf{u}^{\circ}(k+i-2) \cdots \mathbf{u}^{\circ}(k+i-1-sl)\}$ for $i = 1, N$.

Remark 6.2 The observability matrix $\mathbf{\Gamma}_s$ and the lower triangular block Topeliz matrix $\mathbf{H}_s$ in Eq. 6.17 are exactly the same as those in the CT model given by Eq. 6.14, showing that although the numerical integrator transforms the CT signals into discrete data, it preserves the original CT system model in the DT domain. This is in contrast to other CT identification approaches (Johansson 1994, Chou *et al.* 1999), which transform the CT signals and result in a different DT model at the same time.

Remark 6.3 The matrix $\mathbf{X}_{k,N}$ is not measurable. As will be shown later, one does not have to know it in order to identify the models considered here.

Remark 6.4 Similar to $\mathbf{U}^{\circ}_{k,s,N}$, $\mathbf{E}_{k,s,N} \in \Re^{m_s \times N}$ has $s+1$ row blocks and N columns. To obtain consistent estimates of $\mathbf{\Gamma}_s$ and $\mathbf{H}_s$, one has to remove the effect of $\mathbf{E}_{k,s,N}$ on the identification.

Remark 6.5 With the identified $\mathbf{P}_s$, one can generate the PRV for fault detection. From Eqs. 6.6 and 6.16, the PRV is

$$\boldsymbol{\varepsilon}_s(k) = \mathbf{P}_s \begin{bmatrix} \phi_s\left(q^{-1}\right)\mathbf{y}^{\circ}(k) \\ \phi_s\left(q^{-1}\right)\mathbf{u}^{\circ}(k) \end{bmatrix} = \begin{cases} \mathbf{W}_0\mathbf{E}_s(k), & \text{no fault.} \\ \mathbf{W}_0\mathbf{E}_s(k) + \mathbf{W}_0\mathbf{G}_s\boldsymbol{\Xi}_{s,d_i}\boldsymbol{\zeta}_s(\mathbf{f}_{s,d_i}(k)) & \text{with fault.} \end{cases} \tag{6.18}$$

where on the RHS, (i) the first and the second lines are the computational and the internal forms of the PRV, respectively; and (ii) $\boldsymbol{\zeta}_s\left(\mathbf{f}_{s,d_i}(k)\right)$ is the s^{th} multiple integral of $\mathbf{f}_{s,d_i}(t)$ with t ranging from $kT_s - lT_s$ to kT_s, i.e.

$$\boldsymbol{\zeta}_s\left(\mathbf{f}_{s,d_i}(k)\right) = \int_{kT_s-lT_s}^{kT_s}\int_{\tau_1-lT_s}^{\tau_1}\cdots\int_{\tau_{s-1}-lT_s}^{\tau_{s-1}} \mathbf{f}_{s,d_i}(\tau)d\tau_s\ldots d\tau_1$$

Remark 6.6 The fault-free value of $\boldsymbol{\varepsilon}_s(k)$ and the fault-contributed term are represented by $\boldsymbol{\varepsilon}_s^*(k) \equiv \mathbf{W}_0\mathbf{E}_s(k)$ and $\boldsymbol{\varepsilon}_s^f(k) \equiv \mathbf{W}_0\mathbf{G}_s\boldsymbol{\Xi}_{s,d_i}\boldsymbol{\zeta}_s\left(\mathbf{f}_{s,d_i}(k)\right)$, respectively. Apparently $\boldsymbol{\varepsilon}_s^*(k)$ is also a moving-average process of the noise vector $\left[\mathbf{v}^T(k)\ \ \mathbf{o}^T(k)\right]^T$, and follows a zero mean multivariate Gaussian distribution, due to the assumed distribution of $\left[\mathbf{v}^T(k)\ \ \mathbf{o}^T(k)\right]^T$. Consequently, it turns out from Eq. 6.18 that

$$\boldsymbol{\varepsilon}_s(k) = \begin{cases} \boldsymbol{\varepsilon}_s^*(k) \sim N\left(0, \mathbf{R}_{\varepsilon,s}\right), & \text{no fault.} \\ \boldsymbol{\varepsilon}_s^*(k) + \boldsymbol{\varepsilon}_s^f(k) \sim N\left(\boldsymbol{\varepsilon}_s^f(k), \mathbf{R}_{\varepsilon,s}\right), & \text{with fault.} \end{cases}$$

where $\mathbf{R}_{\varepsilon,s}$ is the covariance of $\boldsymbol{\varepsilon}_s^*(k)$. From Eq. 6.16,

$$\mathbf{R}_{\varepsilon,s} = \mathbf{W}_0\left(\mathbf{H}_s\mathbf{Cov}\left(\phi_s\left(q^{-1}\right)\mathbf{v}(k)\right)\mathbf{H}_s^T + \mathbf{Cov}\left(\phi_s\left(q^{-1}\right)\mathbf{o}(k)\right)\right)\mathbf{W}_0^T$$

where, **Cov**( ) is the covariance of the argument. Therefore, one can perform fault detection by simply checking if $\boldsymbol{\varepsilon}_s(k)$ is zero mean. A chi-square distributed variable $\eta_{s,o}(k) = \boldsymbol{\varepsilon}_s^T(k)\mathbf{R}_{\varepsilon,s}^{-1}\boldsymbol{\varepsilon}_s(k)$ is defined. If $\eta_{s,o}(k) > \chi_\alpha^2(ns)$ with α being the selected level of significance, e.g., $\alpha = 0.01$, it indicates that a fault has occurred.

Remark 6.7 The fault gain matrix in $\boldsymbol{\varepsilon}_s(k)$ is $\mathbf{W}_0\mathbf{G}_s$, depending on which a set of SRMs can be designed for fault isolation.

6.5.2 Consistent Estimation of $\boldsymbol{\Gamma}_s$

A choice of $k = 1$ in Eq. 6.17 gives us,

$$\mathbf{Y}_{1,s,N}^{\circ} = \boldsymbol{\Gamma}_s\mathbf{X}_{1,N} + \mathbf{H}_s\mathbf{U}_{1,s,N}^{\circ} + \mathbf{E}_{1,s,N} \tag{6.19}$$

Post-multiplying Eq. 6.19 by $\frac{1}{N}\left[\mathbf{U}^{\circ T}_{L,s,N}\ \mathbf{Y}^{\circ T}_{L,s,N}\right]$ gives

$$\begin{aligned}\frac{1}{N}\mathbf{Y}^{\circ}_{1,s,N}\left[\mathbf{U}^{\circ T}_{L,s,N}\ \mathbf{Y}^{\circ T}_{L,s,N}\right] &= \frac{1}{N}\mathbf{\Gamma}_s\mathbf{X}_{1,N}\left[\mathbf{U}^{\circ T}_{L,s,N}\ \mathbf{Y}^{\circ T}_{L,s,N}\right]\\ &+\frac{1}{N}\mathbf{H}_s\mathbf{U}^{\circ}_{1,s,N}\left[\mathbf{U}^{\circ T}_{L,s,N}\ \mathbf{Y}^{\circ T}_{L,s,N}\right] \qquad (6.20)\\ &+\frac{1}{N}\mathbf{E}_{1,s,N}\left[\mathbf{U}^{\circ T}_{L,s,N}\ \mathbf{Y}^{\circ T}_{L,s,N}\right]\end{aligned}$$

Since the numerical integrator is an $(sl)^{th}$ order filter, selecting $L = sl + 2$ in Eq. 6.20 makes the last term on the RHS asymptotically vanish as $N \rightarrow \infty$.

Substituting the following QR decomposition

$$\begin{bmatrix}\mathbf{U}^{\circ}_{1,s,N}\\ \mathbf{Y}^{\circ}_{1,s,N}\end{bmatrix}\begin{bmatrix}\mathbf{U}^{\circ T}_{L,s,N} & \mathbf{Y}^{\circ T}_{L,s,N}\end{bmatrix} = \begin{bmatrix}\mathbf{R}_{11} & \mathbf{0}\\ \mathbf{R}_{21} & \mathbf{R}_{22}\end{bmatrix}\begin{bmatrix}\mathbf{Q}_1\\ \mathbf{Q}_2\end{bmatrix}$$

into Eq. 6.20 leads to

$$[\mathbf{I}_{m_s}| -\mathbf{H}_s]\begin{bmatrix}\mathbf{R}_{21}\mathbf{Q}_1+\mathbf{R}_{22}\mathbf{Q}_2\\ \mathbf{R}_{11}\mathbf{Q}_1\end{bmatrix} = \mathbf{\Gamma}_s\mathbf{X}_{1,N}\begin{bmatrix}\mathbf{U}^{\circ T}_{L,s,N} & \mathbf{Y}^{\circ T}_{L,s,N}\end{bmatrix} \qquad (6.21)$$

where

$$\mathbf{Q}_i^T\mathbf{Q}_j = \begin{cases}\mathbf{I}, & i=j,\ \ i,j=1,2.\\ \mathbf{0}, & i\neq j,\ \ i,j=1,2.\end{cases}$$

has been employed with **I** standing for an identity matrix of appropriate dimensions.

Post-multiplying Eq. 6.21 by $\mathbf{Q}_2^T$ results in

$$\mathbf{R}_{22} = \mathbf{\Gamma}_s\mathbf{X}_{1,N}\begin{bmatrix}\mathbf{U}^{\circ T}_{L,s,N} & \mathbf{Y}^{\circ T}_{L,s,N}\end{bmatrix}\mathbf{Q}_2^T$$

Applying singular value decomposition (SVD) to $\mathbf{R}_{22}$,

$$\mathbf{R}_{22} = \mathbf{U}_l\mathbf{\Lambda}\mathbf{V}_r^T \qquad (6.22)$$

Therefore, the first n vectors of $\mathbf{U}_l$ give a consistent estimate of $\mathbf{\Gamma}_s$ (up to a similarity transformation), i.e., $\mathbf{\Gamma}_s = \mathbf{U}_l(:,1:n)$, assuming that $\mathbf{X}_{1,N}\begin{bmatrix}\mathbf{U}^{\circ T}_{L,s,N} & \mathbf{Y}^{\circ T}_{L,s,N}\end{bmatrix}\mathbf{Q}_2^T$ has rank n. The validity of this assumption depends on the validity of the consistency conditions which need further investigation.

Remark 6.8 In this formulation, in Eq. 6.3 we have neglected the effect of disturbances. In the presence of disturbances, the multiple integration performed in Eq. 6.11 will result in a relatively high order moving-average disturbance process. In this case it might be necessary to consider a large value of L in Eq. 6.20 in order to obtain a consistent estimate of $\mathbf{\Gamma}_s$. In addition, in order to minimize the effect of the disturbances on the fault diagnosis result, it might be necessary to use robust procedures such as the ones proposed in Frank and Ding (1997).

6.5.3 Calculation of $\mathbf{W}_0$

To calculate the eigenvectors of $\mathbf{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T$, the problem is expressed as:

$$\mathbf{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T\boldsymbol{\nu}_i = \lambda\boldsymbol{\nu}_i$$

where $\boldsymbol{\nu}_i$ is the i^{th} column of the matrix $\mathbf{W}_0^T$, and λ_i is the associated eigenvalue. Introducing $\boldsymbol{\nu}_i = \mathbf{\Gamma}_s^{\perp}\boldsymbol{\omega}_i$ leads to

$$\mathbf{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T\mathbf{\Gamma}_s^{\perp}\boldsymbol{\omega}_i = \lambda_i\mathbf{\Gamma}_s^{\perp}\boldsymbol{\omega}_i$$

Further, since $\mathbf{\Gamma}_s^{\perp}$ is idempotent,

$$\mathbf{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T\mathbf{\Gamma}_s^{\perp}\mathbf{\Gamma}_s^{\perp}\boldsymbol{\omega}_i = \lambda_i\mathbf{\Gamma}_s^{\perp}\boldsymbol{\omega}_i$$

$$\mathbf{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T\mathbf{\Gamma}_s^{\perp}\boldsymbol{\nu}_i = \lambda_i\boldsymbol{\nu}_i.$$

Hence, $\boldsymbol{\nu}_i$ is the i^{th} eigenvector corresponding to the i^{th} non-zero eigenvalue of $\mathbf{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T\mathbf{\Gamma}_s^{\perp}$.
Eventually, $\mathbf{W}_0^T$ = Eigenvectors associated with all the non-zero eigenvalues of the matrix $\mathbf{\Gamma}_s^{\perp}\mathbf{G}_s\mathbf{G}_s^T\mathbf{\Gamma}_s^{\perp}$.

6.5.4 Identification of $\mathbf{W}_0\mathbf{H}_s$.

With the identified $\mathbf{\Gamma}_s$, one can derive $\mathbf{G}_s$ and calculate $\mathbf{W}_0$ following the steps shown earlier. Further, post-multiplying Eq. 6.21 by $\mathbf{W}_0$ yields

$$[\mathbf{W}_0| - \mathbf{W}_0\mathbf{H}_s]\begin{bmatrix}\mathbf{R}_{21}\mathbf{Q}_1 + \mathbf{R}_{22}\mathbf{Q}_2 \\ \mathbf{R}_{11}\mathbf{Q}_1\end{bmatrix} = 0 \qquad (6.23)$$

where $\mathbf{W}_0\mathbf{\Gamma}_s \equiv \mathbf{0}$ is employed. Post-multiplying Eq. 6.23 by $\mathbf{Q}_1^T$ gives

$$\mathbf{W}_0\mathbf{R}_{21} = \mathbf{W}_0\mathbf{H}_s\mathbf{R}_{11}$$

Consequently

$$\mathbf{W}_0\mathbf{H}_s = \mathbf{W}_0\mathbf{R}_{21}\mathbf{R}_{11}^{+}$$

Eventually, $\mathbf{P}_s = [\mathbf{W}_0| - \mathbf{W}_0\mathbf{H}_s]$.

6.5.5 Consistent Estimation of $\{\mathbf{A},\mathbf{B},\mathbf{C},\mathbf{D}\}$

In many scenarios, e.g., in the design of controllers, knowledge of the system matrices is crucial. Therefore, the identification of $\{\mathbf{A}_T,\mathbf{B}_T,\mathbf{C}_T,\mathbf{D}_T\}$ from $\mathbf{\Gamma}_s$ and $\mathbf{W}_0\mathbf{H}_s$ is discussed here. Note that the identified system matrices $\{\mathbf{A}_T,\mathbf{B}_T,\mathbf{C}_T,\mathbf{D}_T\}$ are in fact the similarity transformations of the original ones, e.g.,

$$\mathbf{A}_T = \mathbf{T}\mathbf{A}\mathbf{T}^{-1},\ \mathbf{B}_T = \mathbf{T}\mathbf{B},\ \mathbf{C}_T = \mathbf{C}\mathbf{T}^{-1},\ \mathbf{D}_T = \mathbf{D}$$

where $\mathbf{T} \in \Re^{n\times n}$ is a non-singular matrix.

The estimate of $\mathbf{C}_T$ is simply equal to the first m rows of $\mathbf{U}_l(:,1:n)$, i.e., $\mathbf{C}_T = \mathbf{U}_l(1:m,1:n)$, where $\mathbf{U}_l$ is given by Eq. 6.22. On the other hand, observing that

$$\mathbf{U}_l(m+1:m_s,1:n) = \begin{bmatrix} \mathbf{C}_T\mathbf{A}_T \\ \vdots \\ \mathbf{C}_T\mathbf{A}_T^{s} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_T \\ \vdots \\ \mathbf{C}_T\mathbf{A}_T^{s-1} \end{bmatrix} \mathbf{A}_T = \mathbf{U}_l(1:m_s-m,1:n)\mathbf{A}_T$$

one arrives at

$$\mathbf{A}_T = \mathbf{U}_l^{+}(1:m_s-m,1:n)\mathbf{U}_l(m+1:m_s,1:n)$$

In order to estimate $\mathbf{B}_T$ and $\mathbf{D}_T$, a series of matrix equations are constructed from $\mathbf{W}_0\mathbf{H}_s$. As illustrated earlier in matrix $\mathbf{H}_s$ all the elements above the main diagonal are zero. As a consequence, for each column block with q columns in $\mathbf{W}_0\mathbf{H}_s$, it can be clearly seen, $\forall\, i = 0, s$, that

$$\begin{aligned} &\mathbf{W}_0(:,mi+1:m_s)\mathbf{H}_s(mi+1:m_s,qi+1:qi+q) \\ &= \mathbf{W}_0(:,mi+1:m_s)\begin{bmatrix} \mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{\Gamma}_{s-1-i}\mathbf{T}^{-1} \end{bmatrix}\begin{bmatrix} \mathbf{D}_T \\ \mathbf{B}_T \end{bmatrix} \\ &= \mathbf{W}_0(:,mi+1:m_s)\begin{bmatrix} \mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_l(1:m_s-mi-m,1:n) \end{bmatrix}\begin{bmatrix} \mathbf{D}_T \\ \mathbf{B}_T \end{bmatrix} \end{aligned} \tag{6.24}$$

where $\mathbf{\Gamma}_{-1} = \mathbf{0}$.

Denoting,

$$\mathbf{N}_i = \mathbf{W}_0(:,mi+1:m_s)\mathbf{H}_s(mi+1:m_s,qi+1:qi+q)$$

and

$$\mathbf{M}_i = \mathbf{W}_0(:,mi+1:m_s)\begin{bmatrix} \mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_l(1:m_s-mi-m,1:n) \end{bmatrix}$$

from Eq. 6.24, the following set of equations are established:

$$\begin{bmatrix} \mathbf{N}_0 \\ \mathbf{N}_1 \\ \vdots \\ \mathbf{N}_s \end{bmatrix} = \begin{bmatrix} \mathbf{M}_0 \\ \mathbf{M}_1 \\ \vdots \\ \mathbf{M}_s \end{bmatrix}\begin{bmatrix} \mathbf{D}_T \\ \mathbf{B}_T \end{bmatrix} \tag{6.25}$$

Therefore, the ordinary least squares (OLS) solution to $\{\mathbf{B}_T, \mathbf{D}_T\}$ is (Verhaegen 1994):

$$\begin{bmatrix} \mathbf{D}_T \\ \mathbf{B}_T \end{bmatrix} = \left[\sum_{j=0}^{s} \mathbf{M}_j^T \mathbf{M}_j\right]^{-1} \left[\sum_{j=0}^{s} \mathbf{M}_j^T \mathbf{N}_j\right]$$

6.6 Design of the SRMs for Fault Isolation

With the identified $\mathbf{P}_s$, the PRV required for fault detection can be generated. Further, to isolate faults, the PRV can be transformed into a set of structured residual vectors (SRVs) (Li and Shah 2002), where one SRV is designed to be immune to a specified subset of faults, but has maximized sensitivity to the remaining faults. The performance of the vector-based fault isolation is much better than the performance of the scalar residual based isolation (Li and Shah 2002). Here the SRVs are extended for the isolation of process faults in the CT systems. To generate SRVs, a set of SRMs are designed. This includes the selection of an incidence matrix to characterize the SRVs. The incidence matrix is dependent on the number of faults to be isolated, the system order n and the order of the parity space, and is not unique. A detailed discussion with respect to the selection can be found in (Gertler and Singer 1990, Li and Shah 2002).

In the system under consideration given in Eq. 6.2, the dimension of the fault magnitude vector $\mathbf{f}_{d_i}(t)$ can be up to n, i.e., $1 \leq d_i \leq n$. In the simplest case, only a single fault needs to be isolated. However, in the most difficult case, it might be necessary to isolate up to n multiple faults, although the probability for multiple faults to occur simultaneously is small. Hence there are $\sum_{i=0}^{n} C_i^n = \sum_{i=0}^{n} \frac{n!}{i!(n-i)!}$ fault scenarios in total, where C_i^n is the combination of i from n and $i!$ is the factorial. To isolate these scenarios, an ideal design is to transform the PRV into n SRVs, where the i^{th} SRV is affected with highest sensitivity only by the i^{th} fault, while it is immune to all the other faults, $\forall\ i \in [1, n]$. As will be analyzed later, such an ideal design can be achieved provided that certain conditions are satisfied.

Computationally, the i^{th} SRV is equal to

$$\mathbf{r}_{s,i}(k) = \mathbf{W}_i \boldsymbol{\varepsilon}_s(k) = \mathbf{W}_i \mathbf{P}_s \begin{bmatrix} \mathbf{y}_s^\circ(k) \\ \mathbf{u}_s^\circ(k) \end{bmatrix} = \mathbf{W}_i \left(\boldsymbol{\varepsilon}_s^*(k) + \boldsymbol{\varepsilon}_s^f(k)\right) \tag{6.26}$$

where $\mathbf{W}_i\mathbf{P}_s$ is defined as the i^{th} SRM. Denote $\mathbf{W}_0\mathbf{G}_s \in \Re^{(sm)\times n_s}$ by $\tilde{\mathbf{G}}_s$.

$\boldsymbol{\varepsilon}_s^f(k) = \mathbf{W}_0\mathbf{G}_s\Xi_{s,d_i}\zeta_s\left(\mathbf{f}_{s,d_i}(k)\right)$.

Hence, $\mathbf{r}_{s,i}^f(k) = \mathbf{W}_i\boldsymbol{\varepsilon}_s^f(k) = \mathbf{W}_i\tilde{\mathbf{G}}_s\Xi_{s,d_i}\zeta_s\left(\mathbf{f}_{s,d_i}(k)\right)$. This is the fault-contributed term in $\mathbf{r}_{s,i}(k)$. Note that the last n columns of $\tilde{\mathbf{G}}_s$ are zeros, $\tilde{\mathbf{G}}_s$ has ns non-zero columns.

Apparently, if $\mathbf{r}_{s,i}(k)$ is designed to be insensitive to all but the i^{th} fault $f_i(t)$, $\mathbf{W}_i$ should be orthogonal to the $(n-1)s$ columns associated with $[f_1(t) \ldots f_{i-1}(t)\ f_{i+1}(t) \ldots f_n(t)]^T$ in $\tilde{\mathbf{G}}_s$, $\forall\, i \in [1, n]$.
Denote

$$\tilde{\mathbf{G}}_{s,i} \equiv [\tilde{\mathbf{G}}_s(:, i) \mid \tilde{\mathbf{G}}_s(:, n+i), \ldots, \tilde{\mathbf{G}}_s(:, (s-1)n+i)] \in \Re^{(sn)\times s}, \ \forall\, i = 1, n$$

and the remaining columns in $\tilde{\mathbf{G}}_s$ after leaving out $\tilde{\mathbf{G}}_{s,i}$ by $\tilde{\mathbf{G}}_{s,i}^- \in \Re^{sn\times((n-1)s)}$, respectively. It is desired that $\mathbf{W}_i$ have maximized covariance with $\tilde{\mathbf{G}}_s$ under the constraint $\mathbf{W}_i\tilde{\mathbf{G}}_{s,i}^- = 0$. Since $\tilde{\mathbf{G}}_{s,i}^- \in \Re^{ns\times((n-1)s)}$, $\mathbf{W}_i$ will have $sn - (n-1)s = s$ independent rows. Similarly, it can be shown that, $\forall\, i \in [1, n]$:

$$\mathbf{W}_i^T = \textit{Eigenvectors corresponding to the n non-zero eigenvalues of } \tilde{\mathbf{G}}_{s,i}^{-,\perp}\tilde{\mathbf{G}}_s\tilde{\mathbf{G}}_s^T$$

where, $\tilde{\mathbf{G}}_{s,i}^{-,\perp} = \mathbf{I}_{sn} - \tilde{\mathbf{G}}_{s,i}^-(\tilde{\mathbf{G}}_{s,i}^-)^+ \in \Re^{(sn)\times(sn)}$.

Eq. 6.26 can be further expressed as

$$\mathbf{r}_{s,i}(k) = \mathbf{r}_{s,i}^*(k) + \mathbf{r}_{s,i}^f(k) \in \Re^n \tag{6.27}$$

where, $\mathbf{r}_{s,i}^* = \mathbf{W}_i\boldsymbol{\varepsilon}_s^*(k)$ follows a zero mean Gaussian distribution with covariance $\mathbf{R}_{\varepsilon,s}^i = \mathbf{W}_i\mathbf{R}_{\varepsilon,s}\mathbf{W}_i^T$. Hence $f_i(t) \neq 0$ can be detected and uniquely isolated by checking if $\mathbf{r}_{s,i}(k)$ is non-zero mean. Moreover, the following chi-square distributed random variable can be defined: $\eta_{s,i}(k) = \mathbf{r}_{s,i}^T(k)\left(\mathbf{R}_{\varepsilon,s}^i\right)^{-1}\mathbf{r}_{s,i}(k)$. Then $\eta_{s,i}(k) > \chi_\alpha^2(n)$ indicates the presence of $|f_i(t)| \neq 0$, while $\eta_{s,i}(k) < \chi_\alpha^2(n)$ indicates $|f_i(t)| = 0$. In addition multiple faults can be isolated, e.g., $\{f_i(t) \neq 0\} \cap \{f_j(t) \neq 0\}$, by checking whether $\eta_{s,i}(k)$ and $\eta_{s,j}(k)$ are greater that $\chi_\alpha^2(n)$ simultaneously.

For the sake of easy reference, Table 6.1 gives the incidence matrix to describe how the n SRVs are correlated with the n faults $[f_1(t), \ldots, f_n(t)]^T$. In this table, a "0" indicates the insensitivity of an SRV to a fault, while a "1" indicates that the SRV has highest sensitivity to the fault.

Eventually, given a set of training data the complete procedure for the identification of the PRM and the SRMs for FDI is summarized as follows:

1. Construct $\left\{\mathbf{Y}_{1,s,N}^\circ, \mathbf{U}_{1,s,N}^\circ\right\}$ and $\left\{\mathbf{Y}_{L,s,N}^\circ, \mathbf{U}_{L,s,N}^\circ\right\}$ with $L = sl + 2$.
2. Identify $\boldsymbol{\Gamma}_s$. Derive $\mathbf{G}_s$ from $\boldsymbol{\Gamma}_s$ and calculate $\mathbf{W}_0$. Identify $\mathbf{W}_0\mathbf{H}_s$. Then obtain $\mathbf{P}_s = [\mathbf{W}_0| - \mathbf{W}_0\mathbf{H}_s]$.
3. Select an incidence matrix as illustrated in Table 6.1 to characterize the SRVs. Subsequently calculate $\mathbf{W}_i$, $\forall\, i \in [1, n]$. Consequently, the i^{th} SRM is $\mathbf{W}_i\mathbf{P}_s$.

	$f_1(t)$	$f_2(t)$	$f_3(t)$	$\dots$	$f_n(t)$
$\mathbf{r}_{s,1}(k)$	1	0	0	$\dots$	0
$\mathbf{r}_{s,2}(k)$	0	1	0	$\dots$	0
$\mathbf{r}_{s,3}(k)$	0	0	1	$\ddots$	0
$\mathbf{r}_{s,4}(k)$	0	0	0	$\ddots$	0
$\vdots$	$\vdots$	$\ddots$	$\ddots$	$\ddots$	$\vdots$
$\mathbf{r}_{s,n}(k)$	0	$\dots$	$\dots$	$\dots$	1

Table 6.1: *Incidence matrix to characterize the isolation logic*

6.7 Numerical Results

In this section, the quadruple-tank system (Johansson 2000) illustrated in Fig. 6.1 is used to demonstrate the effectiveness of the proposed approach for the identification of FDI-oriented residual models. Further, to make a comparison DT residual models are also used to carry out detection and isolation of leaks in the tanks. As will be shown, the CT residual models are much more powerful than their DT counterparts at isolating fast time-varying faults.

The parameter values and the chosen operating point of the laboratory process are given in Table 6.2:

A_1, A_3	$[cm^2]$	28
A_2, A_4	$[cm^2]$	32
a_1, a_3	$[cm^2]$	0.071
a_2, a_4	$[cm^2]$	0.057
k_1, k_2	$[cm^3/Vs]$	3.33,3.35
h_1°, h_2°	$[cm]$	12.4,12.7
h_3°, h_4°	$[cm]$	1.8,1.4
v_1°, v_2°	$[V]$	3.00
k_c	$[V/cm]$	0.50
g	$[cm/s^2]$	981
γ_1, γ_2		0.70,0.60

Table 6.2: *Parameter values and chosen operating point of the Quadruple-tank process*

Consequently linearizing Eq. 6.1 under these conditions gives the following CT state space equations (Johansson 2000):

$$
\begin{aligned}
\dot{\mathbf{x}}(t) &= \begin{bmatrix} -0.0159 & 0 & 0.0419 & 0 \\ 0 & -0.0111 & 0 & 0.0333 \\ 0 & 0 & -0.0419 & 0 \\ 0 & 0 & 0 & -0.0333 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0.0833 & 0 \\ 0 & 0.0628 \\ 0 & 0.0479 \\ 0.0312 & 0 \end{bmatrix} \mathbf{u}(t) \\
\mathbf{y}(t) &= \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix} \mathbf{x}(t)
\end{aligned}
\tag{6.28}
$$

The inputs to the system are simulated by pseudo random binary signals with a small magnitude. The frequency band for the frequency contents of the inputs is chosen to be $[0, 0.03]$, expressed as a fraction of the Nyquist frequency (see the *idinput* command in MATLAB©). Using SIMULINK© directly in the CT domain, training data is generated for model identification and test data for FDI. In this simulation the CT input and output signals are sampled with a sampling interval T_s. Further, an independent identically distributed noise vector with gaussian density characterized by the covariance matrix $0.1^2\mathbf{I}_6$, i.e., $\aleph(0, 0.1^2\mathbf{I}_6)$ is generated and introduced to the discrete noise-free inputs and outputs, resulting in a set of noise-contaminated training data with 5000 samples. Subsequently, we compute $\mathbf{P}_s$.

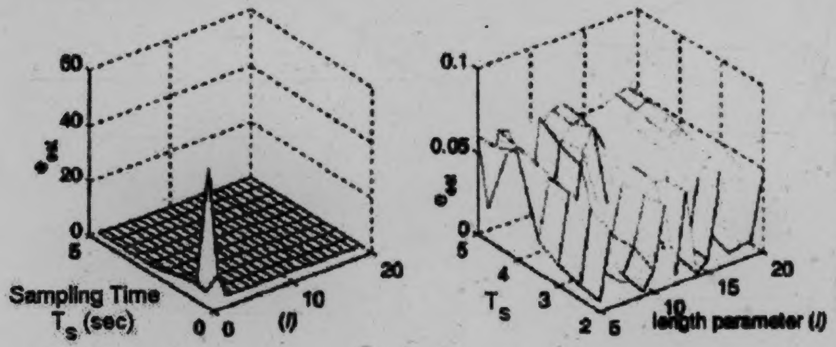


Figure 6.2: Selection of T_s and l for continuous time system

In the CT SIM, two important parameters: the sampling time, T_s and the length parameter of the integral filter l need to be determined. The effect of T_s and l on the estimation of the CT model parameters is shown in Fig. 6.2, where the estimation error is defined by:

$$e_{est} = \|eig(A_{est}) - eig(A)\|_2$$

with $eig(\,)$ standing for a vector containing all the eigenvalues of a matrix in a descending order and $\|\ \|_2$ standing for the 2-norm of a vector. T_s and l are chosen to be equal to 2 sec and 14, respectively in this case.

With the chosen l and T_s, using the training data $\mathbf{\Gamma}_s \in \Re^{20\times4}$ is identified, from which $\mathbf{G}_s \in \Re^{20\times20}$ is derived. Further $\mathbf{W}_0 \in \Re^{16\times20}$ is calculated and $\mathbf{W}_0\mathbf{H}_s \in \Re^{16\times10}$ identified, resulting in $\mathbf{P}_s = [\mathbf{W}_0| - \mathbf{W}_0\mathbf{H}_s] \in \Re^{16\times30}$. $\mathbf{R}_{\varepsilon,s} \in \Re^{16\times16}$ is estimated and the four transformation matrices $\{\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{W}_4\}$ are calculated. Eventually the four SRMs: $\mathbf{W}_i\mathbf{P}_s \in \Re^{4\times30}$ and four covariances, $\mathbf{R}^i_{\varepsilon,s} \in \Re^{4\times4}\ \forall\, i \in [1,4]$ are obtained.

Faults are introduced as leaks in the tanks. In accordance with the laws of fluid mechanics, a leak is proportional to the square root of the water level in the tank, and is time-varying, i.e.,

$$\delta_i(t) = a_i^f\sqrt{2gh_i(t)},\ \forall\, i = 1,4$$

where, a_i^f is the size of the leak orifice. Clearly such a fault is not piece-wise constant because $h_i(t)$ can be very time-varying within one sampling interval.

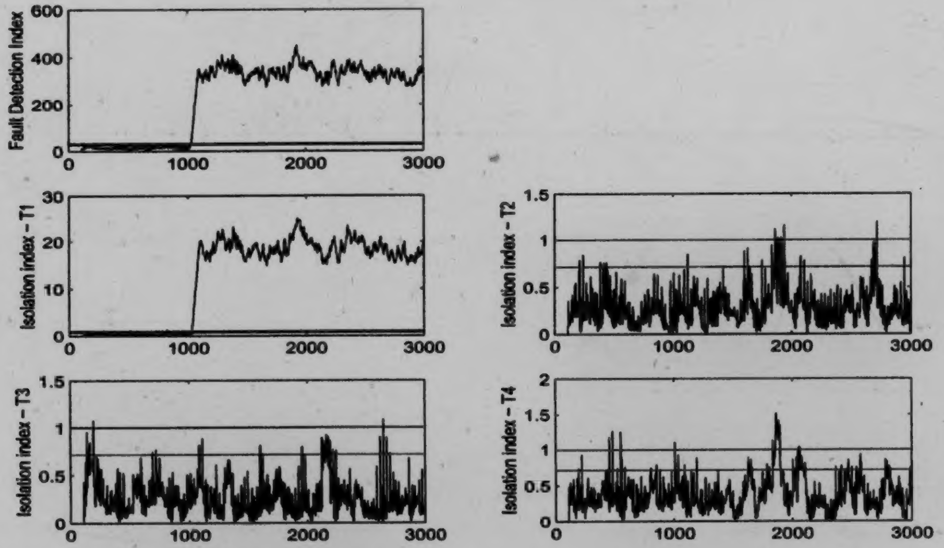


Figure 6.3: CT residual model-based detection and isolation of a leak in Tank 1

FDI is conducted in the following four cases:

Case 1: A leak simulated by selecting $a_i^f = 0.01\ cm^2$ is introduced to a single tank after the 1000th second and the relevant FDI results are shown in Fig. 6.3, where for the

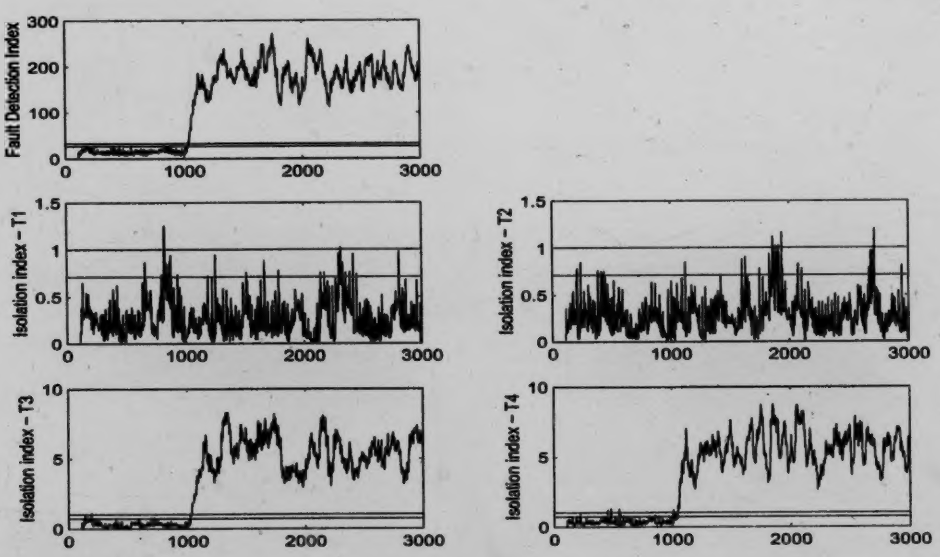


Figure 6.4: CT residual model-based detection and isolation of two simultaneous leaks in Tanks 3 and 4

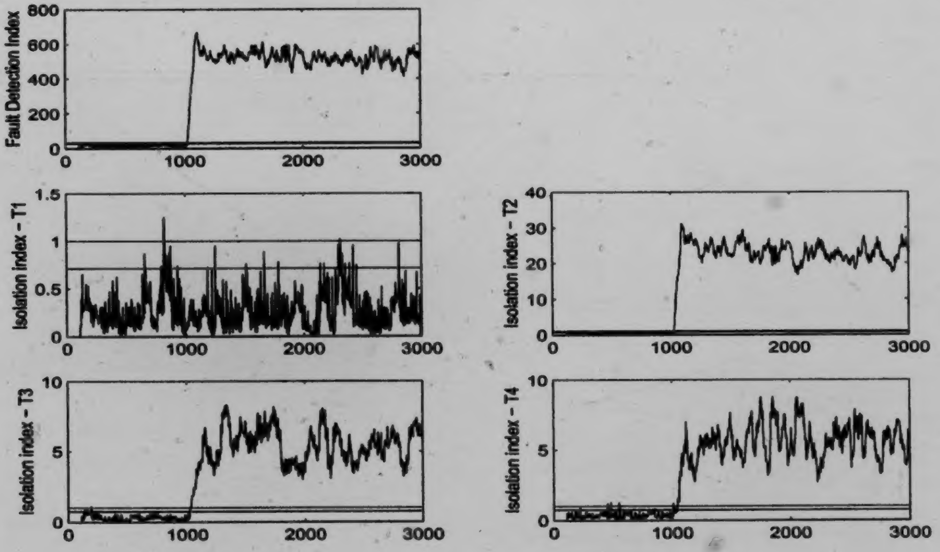


Figure 6.5: CT residual model-based detection and isolation of three simultaneous leaks in Tanks 2, 3 and 4

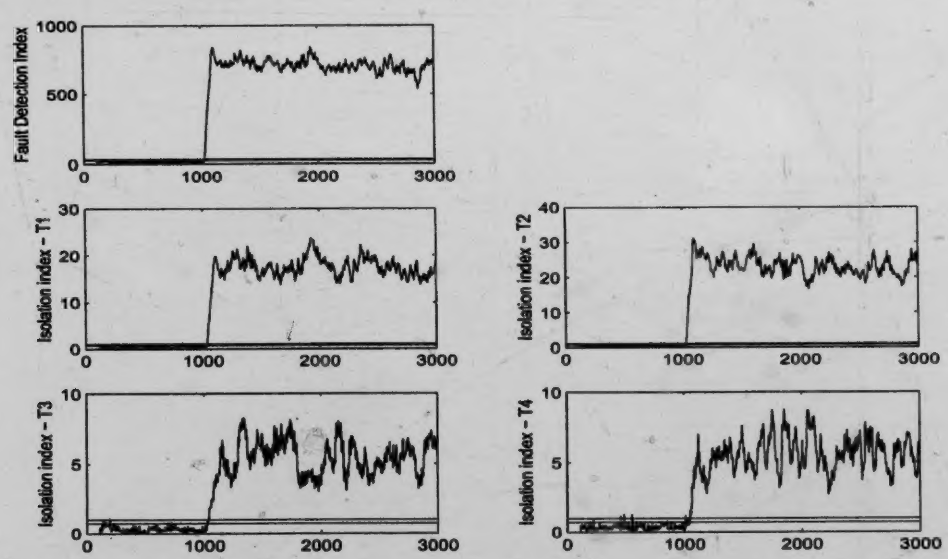


Figure 6.6: CT residual model-based detection and isolation of simultaneous leaks in all 4 Tanks

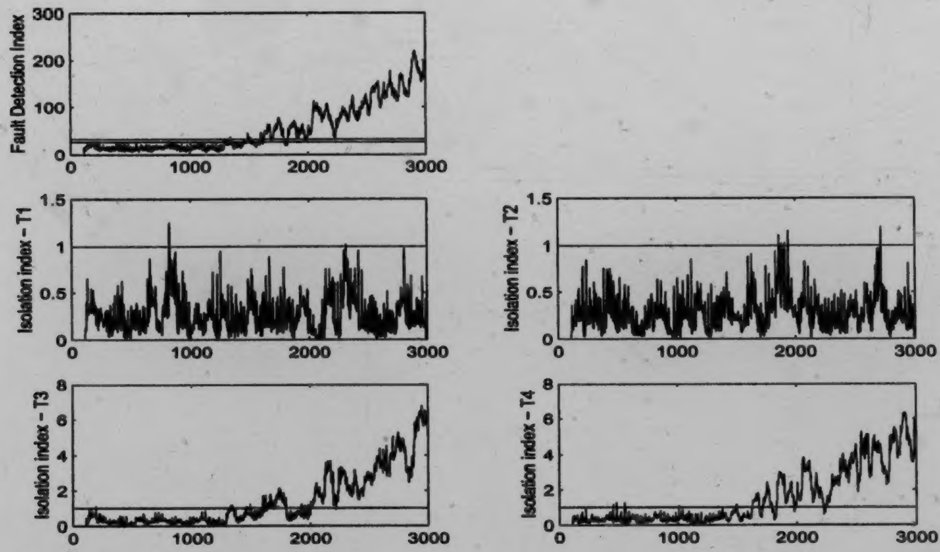


Figure 6.7: CT residual model-based detection and isolation of two simultaneous incipient leaks in tanks 3 and 4.

sake of better visualization, all the isolation indices have been scaled to have unit confidence limit, i.e., $\eta^{\circ}_{s.i}(k) = \frac{\eta_{s.i}(k)}{\chi^2_{\alpha}(4)|_{\alpha=0.01}}$. Note that in the following cases, all the fault isolation indices have also been scaled to have unit confidence limits. Since $\eta^{\circ}_{s.1}(k)$ is greater than its confidence limit 1, but $\eta^{\circ}_{s.i}(k)$, $\forall\, i \in [2,4]$, are all less than their confidence limits, it can be inferred that Tank 1 is leaking.

Case 2: Two leaks simulated with the choice of $a^f_i = a^f_j = 0.01\ cm^2$ are introduced to two different tanks simultaneously after the 1000^{th} second. As shown in Fig. 6.4, the faults are detected promptly. Furthermore, since $\eta^{\circ}_{s.3}(k)$ and $\eta^{\circ}_{s.4}(k)$ are greater than their confidence limits 1, but $\eta^{\circ}_{s.1}(k)$ and $\eta^{\circ}_{s.2}(k)$ are lower than their confidence limits, it is indicated that Tanks 3 and 4 are leaking at the same time.

Case 3: Three leaks with $a^f_i = a^f_j = a^f_k = 0.01\ cm^2\ \forall\, i,j,k \in [1,4]$ are introduced to three different tanks simultaneously after the 1000^{th} second and the FDI results are shown in Fig. 6.5. After the detection of faults, since $\eta^{\circ}_{s.2}(k)$, $\eta^{\circ}_{s.3}(k)$ and $\eta^{\circ}_{s.4}(k)$ are greater than their confidence limits 1 simultaneously, but $\eta^{\circ}_{s.1}(k)$ is lower than its confidence limit, it can be concluded that Tanks 2, 3 and 4 are leaking.

Case 4: Leaks with $a^f_i = a^f_j = a^f_k = a^f_\tau = 0.01\ cm^2$, $\forall\, i,j,k,\tau \in [1,4]$ are introduced to all four tanks simultaneously after the 1000^{th} second. As Fig. 6.6 shows, the faults are detected promptly. Furthermore, since $\eta^{\circ}_{s.i}(k)$, $\forall\, i = 1,2,3,4$, are all greater than their confidence limits 1, it is known that all the tanks are leaking.

Case 5: Two leaks simulated with the choice of $a^f_{i1} = a^f_{i2} = 0.01 \times (t-1000)/2000cm^2$ are introduced to two different tanks simultaneously after the 1000^{th} second. In this case, note that the magnitude of a leak increases slowly. The detection of such an incipient fault is generally regarded as difficult. As shown in Fig. 6.7, the proposed scheme is still able to detect the incipient faults after these faults have developed over a period of time. Furthermore, it is clearly indicated that tanks 3 and 4 are leaking simultaneously.

Mathematically, the PRM and the fault gain matrix in the DT domain can be derived from their counterparts in the CT domain by replacing the CT system matrices as follows:

To make a comparison, the DT model is used to design the PRM and SRMs. With $T_s = 2$, $\mathbf{A}_d = \mathbf{e}^{\mathbf{A}T_s}$, $\mathbf{B}_d = \int_0^{T_s} \mathbf{e}^{\mathbf{A}\tau}\mathbf{B}d\tau$, $\mathbf{C}_d = \mathbf{C}$, $\mathbf{D}_d = \mathbf{D}$ are obtained. The fault model is $\mathbf{e}^{\mathbf{A}T_s} \int_0^{T_s} \mathbf{e}^{-\mathbf{A}\tau}\Xi_{d_i} d\tau$ assuming piece-wise constancy of the faults, where the true values of $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and $\mathbf{D}$ are used. Subsequently, the DT PRM and four SRMs are computed by simply replacing $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ with $\{\mathbf{A}_d\mathbf{B}_d\mathbf{C}_d\mathbf{D}_d\}$, respectively.

CT Domain	$\mathbf{A}$	$\mathbf{B}$	$\mathbf{C}$	$\mathbf{D}$	Ξ_{d_i}
DT Domain	$\mathbf{A}_d$	$\mathbf{B}_d$	$\mathbf{C}_d$	$\mathbf{D}_d$	$\mathbf{e}^{\mathbf{A}T_s}\int_0^{T_s}\mathbf{e}^{-\mathbf{A}\tau}d\tau\Xi_{d_i}$

Table 6.3: *Correspondences between the PRM and the fault gain matrix in the CT domain and their counterparts in the DT domain*

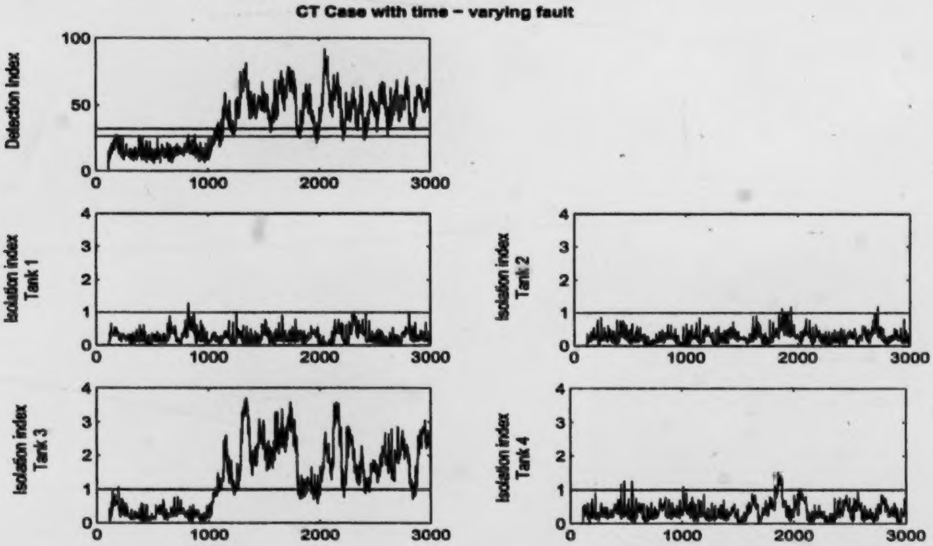


Figure 6.8: The CT residual model-based FDI scheme successfully detects and isolates a time-varying leak in Tank 3

A single leak to a tank is introduced followed by the application of the CT and DT residual models to the same test data. It is shown in Figs. 6.8 and 6.9 that while the identified CT residual models successfully detect and isolate a leak in Tank 3, the DT residual models fail to detect and isolate the time-varying fault even if they are calculated from the exactly known **A**, **B**, **C** and **D**. Following this a ZOH is introduced to the fault, assuming that the fault is piece-wise constant within each sampling interval. As depicted in Fig. 6.10, in this case, the DT-FDI performance has improved. However, it is still much worse than the CT-FDI performance shown in Fig. 6.8.

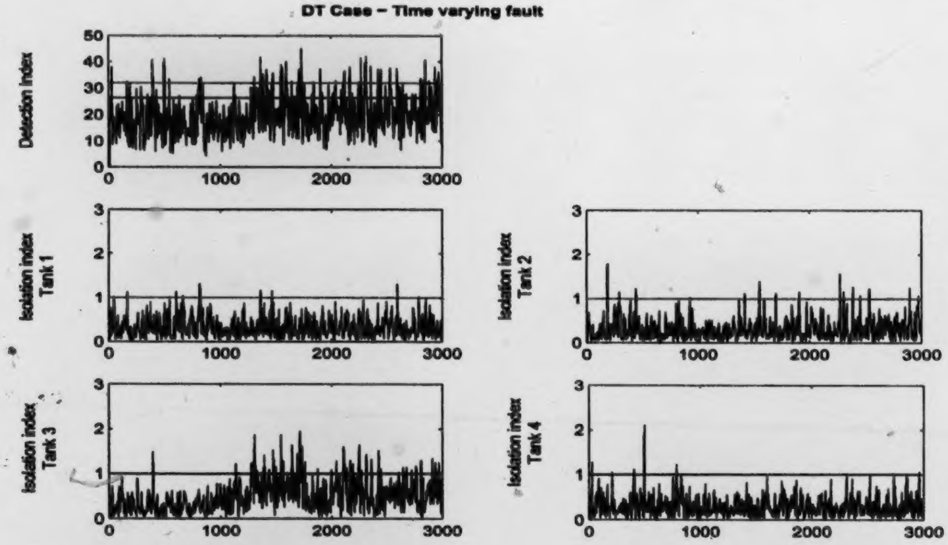


Figure 6.9: The DT residual model-based FDI scheme fails to detect and isolate a time-varying leak in Tank 3

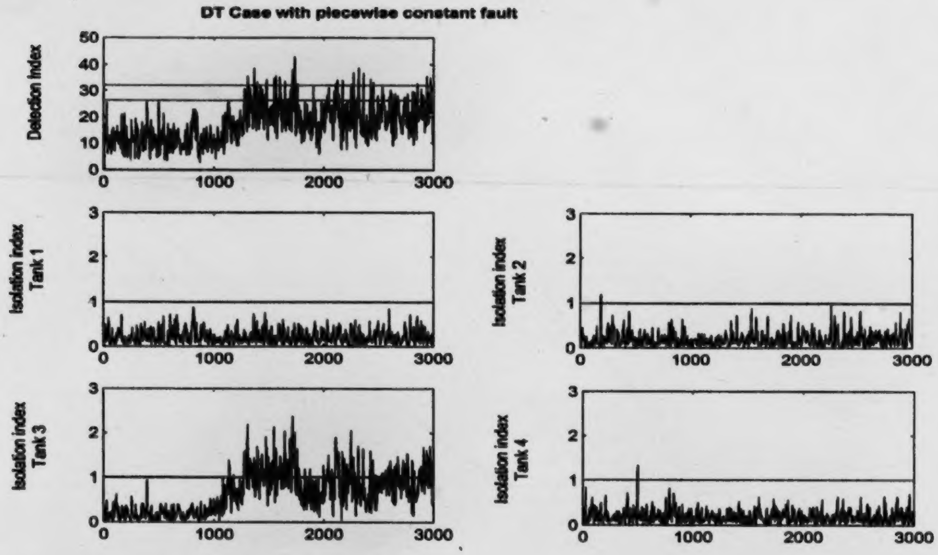


Figure 6.10: The DT residual models can detect and isolate a leak with a ZOH in Tank 3 , with relatively poor performance

6.8 Conclusions

In this chapter, a novel subspace approach to the identification of the PRM for fault detection and the design of a set of SRMs for fault isolation in multivariate CT systems has been proposed. When the number of outputs is equal to the system order, single and multiple faults can be uniquely isolated by the use of n SRVs, where one SRV is made affected only by a single fault but unaffected by the other faults. The proposed scheme is computationally robust and efficient, without the need to separately identify the system matrices $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$. The newly proposed approach is applied to a simulated water tank system, where detection and isolation of leaks in a single tank at one time and in multiple tanks simultaneously has been successfully conducted. In addition, to make a comparison, the results of detecting and isolating leaks in the tanks using DT PRM and SRMs are also shown. It is shown that the DT-FDI approach can just barely detect and identify only the piece-wise constant fault. In contrast the CT-FDI scheme works well for detecting and isolating single as well as multiple leaks from any tank.

7
Conclusions

7.1 Summary

This thesis has presented the use of quantitative techniques for fault detection and diagnosis in process industries. The thesis has confirmed the applicability of existing data-driven and model-based techniques and introduced a number of novel techniques for model identification, fault detection and diagnosis.

A simulation case-study was presented in Chapter 1 to demonstrate the importance of FDD especially in closed-loop systems. Two industrial case-studies were presented in Chapter 2 to demonstrate the applicability of multivariate statistical techniques for process monitoring.

The results of the PCA-based failure diagnosis are being used in an early-warning scheme to prevent bearing damage in the turbine in a thermal power plant in Missouri, USA. This early-warning scheme is used to send alerts to process engineers through electronic mail when large shifts in the turbine bearing temperatures or the turbine-shaft eccentricity is detected. The inferential sensor-based bitumen recovery monitoring scheme is being used in Suncor's oil sands extraction facility in Fort McMurray, AB, Canada, to supplement laboratory analysis and provide advance warning of an impending fall in the operating efficiency of the separation cell. This study also helped improve our understanding of some of the practical issues in developing steady-state models for processes from archived, historical data. In particular, understanding the effects of settings in the data historian and the effect of sample consolidation and the role of nonlinear transformation were found to be very important in improving the quality of the model. In addition, this application helped us develop new techniques for time-delay estimation and test the applicability of a powerful new iterative PCA technique. The simulation studies on the quadruple tank process have resulted in the development of optimal choices for the residual generation mechanism and decision rule criteria which result in maximized sensitivity of the SRV approach.

In Chapter 3, we have demonstrated, using laboratory experiments and simulation case-studies, that the SRV approach can be applied successfully to detect and isolate sensor and actuator faults under closed-loop conditions. We have also demonstrated that this approach is more powerful than the contribution plots associated with PCA for the diagnosis of sensor and actuator faults.

In practice, we have come across a number of situations where a model is not readily available. The bitumen separation cell in Suncor's oil sands extraction facility and the bleaching unit in Millar Western's BCTMP mill are two examples. There is ongoing effort in the engineering community to understand these processes and develop phenomenological models using physical principles. However, the development of such models requires a significant investment of time and money. In the meantime, for a number of applications such as, control and fault diagnosis, data-based empirical models are required. It is difficult to apply currently existing system identification techniques to these processes mainly because of the multirate nature of these processes. The solutions proposed to this problem, like linear interpolation, are unsatisfactory, because the large difference in the sampling rates falsify the hypothesis involved in such interpolations. When we tried to use the existing techniques to develop models for the BCTMP mill, the model quality was terrible even though the predictions of the quality variables, like brightness and tensile strength, looked good. We soon realized that we were trying to identify the *right* model with the *wrong* data. Indeed there was hardly any justification for assuming that we could draw a straight line, or an n^{th} order polynomial for that matter, between available output samples and expect the unavailable data to closely match this interpolation. On the other hand, valuable information regarding the output was available in the input. Hence, a more intelligent form of interpolation, a model-based interpolation, was required. The next question was, *where do we get the model from?* This sounded like the classic *chicken and egg* problem. We soon discovered that this problem was tackled about 25 years ago in statistical literature, by Dempster *et al.* (1977), in an iterative fashion. After identifying a suitable gray-box model structure which takes the recycle and nonlinear effects into account, we applied the EM algorithm to this structure.

The predictors developed using this model have been functioning successfully for over one-and-a-half years at the mill. The identification technique has also been successfully tested in other simulation and experimental studies.

In Chapter 6, a novel subspace approach for the identification of fault diagnosis relevant models for fault isolation in multivariate continuous-time systems has been proposed. This approach is successfully applied for the detection and isolation of leaks in a simulated water tank system.

7.2 Contributions of this thesis

The key contributions of this study are as follows:

- Two industrial case-studies involving PCA and PLS are presented to demonstrate the applicability of multivariate statistical techniques for process monitoring and to illustrate some of the issues involved in developing inferential sensors from archived historical data.

- A powerful novel approach for the joint identification of steady-state models and the noise covariance matrix (Iterative Principal Components Analysis (IPCA)) was successfully tested in an industrial setting.

- An averaging mechanism to compensate for sample consolidation in multivariate static model identification has been developed. A PEM-based optimal transport-delay estimation technique for multivariate static model identification has been developed.

- The superiority of SRV-based fault isolation over scalar structured residual approaches has been demonstrated.

- Optimal choices for the residual generation mechanism and decision rule criteria have been established using extensive simulation studies on a quadruple tank process.

- The applicability of the SRV-based approach for detection and isolation of sensor and actuator faults under closed-loop conditions has been demonstrated using laboratory experiments and simulation case-studies. The superiority of the SRV-based isolation scheme over contribution plots has also been demonstrated.

- A novel EM-based approach for the identification of multivariate state-space models for processes with irregularly sampled outputs has been developed. Its superiority over ad-hoc interpolation schemes has been demonstrated using simulations, laboratory experiments and industrial case-studies.

- A novel subspace approach for the identification of fault diagnosis relevant models for fault isolation in multivariate continuous-time systems has been developed. This approach is demonstrated through leak detection and isolation in a simulated water tank system.

7.3 Recommendations for future work

The results and issues identified during the course of this work give rise to a number of possible directions for future work. Some of these are briefly summarized below.

- A new static model identification method (IPCA) has been introduced in this thesis. The theoretical properties of this approach and related EIV modelling techniques such as PLS are yet to be established. In particular, the highly convoluted procedure used in the PLS algorithm has to be studied with the objective of establishing its properties.

- Optimal settings for the SRV-based fault isolation method have been recommended. Alternately one can use the Generalized Likelihood Ratio (GLR) approach for fault isolation. Intuitively, the GLR and SRV approaches seem to be perfectly complementary. Hence, by a proper choice the decision rule and residual generation mechanism, they can give exactly the same results. However a theoretical proof for this is lacking and will help establish these methods on an equal footing. In addition, while intuitive arguments establish that parity space approaches are equivalent to observer-based approaches, a more theoretical investigation into the conditions for equivalence have to be established.

- The use of the SRV-based approach for sensor and actuator FDI under closed-loop conditions has been demonstrated. However, the effect of disturbances and model-plant-mismatch has not been considered in this thesis. These are important factors to consider when these methods are used in an online FDD scheme and these effects have to be investigated.

- We have considered fault types, such as sensor, actuator and additive process faults, in isolation and not simultaneously. A theoretical study of the isolability of these faults when they occur simultaneously will help improve the maturity of online fault diagnosis. In particularly, the study of redundancy classes will help in avoiding fault misclassification.

- We have introduced the EM algorithm for identification of state-space models from irregularly sampled data. In addition, it provides an optimal alternative to the existing PEM and Subspace identification methods for multivariate system identification from regularly sampled data. The model parameters estimated using this approach are known to possess maximum likelihood properties asymptotically. However, work on obtaining confidence interval and establishing finite sample properties is yet to be done in this area. In addition, a theoretical comparison of this approach with subspace identification will help in establishing asymptotic properties of various subspace algorithms.

A

Comparison of structured residual fault isolation techniques

A.1 FDI performance tables

The following tables summarize the results of monte-carlo simulations performed to study the performance of structured residual based fault detection and isolation on the quadruple tank process. The performance of 6 different settings corresponding to different choices of the residual generation approach and decision rule are summarized. In each table, each row shows the detection and isolation results for 100000 cases corresponding to different realizations of the noise sequences. Different rows correspond to different faults sizes (0.5σ, 1σ, 2σ, 3σ and 5σ) and fault locations ($F_1, \cdots, F_8$).

Fault Description: No fault			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
N/A	98968	1032	317	4	1	2	3	0	0	0	0	705
Fault Description: Fault in F1			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98929	1071	302	4	3	1	4	1	0	0	0	756
1σ	98557	1443	357	8	2	2	2	4	3	0	0	1065
2σ	96302	3698	674	38	3	6	15	3	6	0	0	2953
3σ	89961	10039	1310	117	1	13	22	6	21	0	0	8549
5σ	55514	44486	2247	992	3	16	47	5	45	0	0	41131
Fault Description: Fault in F2			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98921	1079	291	1	3	2	4	0	1	0	0	777
1σ	98427	1573	358	4	7	1	3	7	1	0	0	1192
2σ	95577	4423	747	5	25	11	6	10	2	0	0	3617
3σ	87843	12157	1288	3	108	18	8	22	0	0	0	10710
5σ	49755	50245	2130	0	869	18	10	36	0	0	0	47182
Fault Description: Fault in F3			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98911	1089	315	2	7	1	4	1	1	0	0	758
1σ	98501	1499	361	6	4	2	3	3	1	0	0	1119
2σ	95875	4125	761	9	12	20	3	3	0	0	0	3317
3σ	88750	11250	1436	13	24	107	2	23	3	0	0	9642
5σ	51781	48219	2537	14	48	909	0	55	2	0	0	44654
Fault Description: Fault in F4			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98967	1033	282	1	1	2	5	2	1	0	0	739
1σ	98455	1545	354	5	2	1	9	5	3	0	0	1166
2σ	95594	4406	759	5	8	2	34	4	4	0	0	3590
3σ	87923	12077	1420	18	10	3	112	5	19	0	0	10490
5σ	50072	49928	2246	24	21	0	882	0	35	0	0	46720

Table A.1: *Results of FDI using scalar structured residual approach with multiple testing - Part1*

Fault Description: Fault in F5			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98836	1164	342	1	2	2	1	3	0	0	0	813
1σ	98047	1953	538	4	3	5	2	7	1	0	0	1393
2σ	93302	6698	1707	2	17	9	0	43	6	0	0	4914
3σ	79683	20317	4352	2	26	21	1	197	4	0	0	15714
5σ	28033	71967	8046	1	32	23	0	1663	6	0	0	62196
Fault Description: Fault in F6			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98862	1138	355	0	1	2	1	0	3	0	0	776
1σ	98175	1825	568	2	1	0	4	3	9	0	0	1238
2σ	93591	6409	1723	9	4	5	16	14	39	0	0	4599
3σ	80310	19690	4453	22	5	1	28	16	185	0	0	14980
5σ	28950	71050	8568	30	0	0	42	4	1626	0	0	60780
Fault Description: Fault in F7			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98886	1114	354	0	2	1	1	1	1	0	0	754
1σ	98201	1799	639	0	1	2	1	2	1	0	0	1153
2σ	93816	6184	2568	3	1	2	4	1	1	0	0	3604
3σ	81358	18642	8325	0	0	4	2	0	0	0	0	10311
5σ	30369	69631	27530	3	0	2	0	0	0	6	0	42090
Fault Description: Fault in F8			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98894	1106	347	0	5	1	1	1	1	0	0	750
1σ	98208	1792	639	0	3	2	1	1	0	0	0	1146
2σ	93690	6310	2637	3	2	0	2	1	0	0	0	3665
3σ	80760	19240	8770	2	1	0	1	0	0	0	0	10466
5σ	29823	70177	28402	0	0	0	1	0	0	0	8	41766

Table A.2: *Results of FDI using scalar structured residual approach with multiple testing - Part2*

Fault Description: No fault			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
N/A	98968	1032	365	4	2	4	5	0	0	0	0	652
Fault Description: Fault in F1			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98929	1071	419	5	4	5	3	0	0	0	0	635
1σ	98557	1443	478	11	4	5	12	0	0	0	0	933
2σ	96302	3698	744	72	5	18	24	0	0	0	0	2835
3σ	89961	10039	1146	253	8	37	39	0	0	0	0	8556
5σ	55514	44486	1145	1934	6	46	54	0	0	0	0	41301
Fault Description: Fault in F2			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98921	1079	401	3	4	3	4	0	0	0	0	664
1σ	98427	1573	483	6	13	5	4	0	0	0	0	1062
2σ	95577	4423	759	6	65	13	16	0	0	0	0	3564
3σ	87843	12157	1119	4	256	30	31	0	0	0	0	10716
5σ	49755	50245	969	3	1803	32	31	0	0	0	0	47407
Fault Description: Fault in F3			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98911	1089	428	5	3	2	5	0	0	0	0	646
1σ	98501	1499	465	6	8	7	1	0	0	0	0	1012
2σ	95875	4125	743	22	25	59	9	0	0	0	0	3267
3σ	88750	11250	1164	37	38	255	6	0	0	0	0	9750
5σ	51781	48219	1069	41	49	1860	2	0	0	0	0	45198
Fault Description: Fault in F4			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98967	1033	393	4	3	5	7	0	0	0	0	621
1σ	98455	1545	491	8	5	7	10	0	0	0	0	1024
2σ	95594	4406	820	13	21	5	73	0	0	0	0	3474
3σ	87923	12077	1260	34	42	7	256	0	0	0	0	10478
5σ	50072	49928	1086	41	60	2	1842	0	0	0	0	46897

Table A.3: *Results of FDI using scalar structured residual approach with maximized sensitivity and multiple testing - Part1*

Fault Description: Fault in F5			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98836	1164	468	3	7	2	3	0	0	0	0	681
1σ	98047	1953	813	2	13	3	3	0	0	0	0	1119
2σ	93302	6698	3078	5	24	10	3	0	0	0	0	3578
3σ	79683	20317	9354	3	46	34	0	0	0	0	0	10880
5σ	28033	71967	25703	1	105	63	2	0	0	0	0	46093
Fault Description: Fault in F6			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98862	1138	472	0	3	2	7	0	0	0	0	654
1σ	98175	1825	804	4	4	1	11	0	0	0	0	1001
2σ	93591	6409	3136	13	5	2	22	0	0	0	0	3231
3σ	80310	19690	9512	35	6	7	49	0	0	0	0	10081
5σ	28950	71050	26933	93	3	0	112	0	0	0	0	43909
Fault Description: Fault in F7			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98886	1114	467	3	4	4	4	0	0	0	0	632
1σ	98201	1799	817	3	8	7	8	0	0	0	0	956
2σ	93816	6184	3093	14	7	20	6	0	0	0	0	3044
3σ	81358	18642	9422	37	8	43	3	0	0	0	0	9129
5σ	30369	69631	27228	105	3	126	1	0	0	0	0	42168
Fault Description: Fault in F8			Number of samples: 100000									
Size	Detection Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98894	1106	437	0	2	2	2	0	0	0	0	663
1σ	98208	1792	743	3	8	2	3	0	0	0	0	1033
2σ	93690	6310	2861	3	22	3	13	0	0	0	0	3408
3σ	80760	19240	8599	1	47	3	39	0	0	0	0	10551
5σ	29823	70177	24230	1	121	0	82	0	0	0	0	45743

Table A.4: *Results of FDI using scalar structured residual approach with maximized sensitivity and multiple testing - Part2*

Fault Description: No fault			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
N/A	98968	1032	0	6	1	0	5	4	3	7	7	999
Fault Description: Fault in F1			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98929	1071	0	6	2	1	6	3	3	5	5	1040
1σ	98557	1443	0	12	3	2	9	7	11	11	7	1381
2σ	96302	3698	0	128	11	11	14	10	26	28	13	3457
3σ	89961	10039	0	766	25	30	24	17	58	63	31	9025
5σ	55514	44486	0	10651	48	58	58	30	118	126	34	33363
Fault Description: Fault in F2			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98921	1079	0	6	5	1	7	6	4	5	10	1035
1σ	98427	1573	0	4	12	5	8	10	6	6	11	1511
2σ	95577	4423	0	11	165	14	13	21	10	15	35	4139
3σ	87843	12157	0	17	956	33	26	48	21	27	62	10967
5σ	49755	50245	0	25	12359	67	48	100	22	30	118	37476
Fault Description: Fault in F3			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98911	1089	0	5	1	3	7	3	4	5	8	1053
1σ	98501	1499	0	5	4	19	8	9	7	9	9	1429
2σ	95875	4125	0	7	8	175	15	28	9	32	16	3835
3σ	88750	11250	0	19	25	926	24	62	16	57	25	10096
5σ	51781	48219	0	44	50	11869	34	115	17	129	35	35926
Fault Description: Fault in F4			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98967	1033	0	7	2	0	8	3	6	5	9	993
1σ	98455	1545	0	7	3	0	21	4	11	4	10	1485
2σ	95594	4406	0	9	7	3	174	15	31	16	23	4128
3σ	87923	12077	0	19	32	10	1011	25	55	23	43	10859
5σ	50072	49928	0	50	59	40	12419	24	108	28	115	37085

Table A.5: *Results of FDI using structured residual vector approach with maximized sensitivity and multiple testing - Part1*

Fault Description: Fault in F5			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98836	1164	0	6	3	1	4	12	5	5	5	1123
1σ	98047	1953	0	7	6	4	5	61	8	10	10	1842
2σ	93302	6698	0	8	32	22	7	572	14	27	23	5993
3σ	79683	20317	0	13	78	60	22	3397	29	47	48	16623
5σ	28033	71967	0	6	132	111	9	31878	30	50	48	39903
Fault Description: Fault in F6			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98862	1138	0	5	1	2	5	6	17	6	9	1087
1σ	98175	1825	0	7	4	3	14	11	62	9	14	1701
2σ	93591	6409	0	23	9	8	32	27	570	25	30	5685
3σ	80310	19690	0	52	16	14	76	37	3344	43	54	16054
5σ	28950	71050	0	110	16	16	149	27	31690	43	70	38929
Fault Description: Fault in F7			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98886	1114	0	6	3	0	7	6	4	16	6	1066
1σ	98201	1799	0	7	4	7	9	9	8	55	10	1690
2σ	93816	6184	0	18	11	33	9	25	29	556	24	5479
3σ	81358	18642	0	44	20	75	16	49	43	3319	33	15043
5σ	30369	69631	0	101	19	127	19	62	46	32224	32	37001
Fault Description: Fault in F8			Number of samples: 100000									
Size	Fault Present?		Isolation									
	No	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8	U
0.5σ	98894	1106	0	5	2	0	6	3	3	7	15	1065
1σ	98208	1792	0	4	6	0	9	9	6	7	67	1684
2σ	93690	6310	0	6	28	13	30	29	24	17	526	5637
3σ	80760	19240	0	14	76	25	63	65	50	33	3299	15615
5σ	29823	70177	0	12	152	15	129	66	54	24	31556	38169

Table A.6: *Results of FDI using structured residual vector approach with maximized sensitivity and multiple testing - Part2*

Fault Description: No fault			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
N/A	1032	98968	153	149	149	130	144	136	78	93
Fault Description: Fault in F1			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1071	98929	148	147	133	144	167	157	75	100
1σ	1443	98557	270	145	194	189	195	220	121	109
2σ	3698	96302	1576	140	614	409	236	640	354	127
3σ	10039	89961	4354	96	1644	824	295	1717	941	138
5σ	44486	55514	28873	36	7350	1460	229	5432	3028	78
Fault Description: Fault in F2			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1079	98921	110	195	134	153	175	132	67	113
1σ	1573	98427	119	391	171	233	274	149	73	163
2σ	4423	95577	96	1092	374	731	802	194	90	444
3σ	12157	87843	70	5889	789	2060	1954	269	112	1034
5σ	50245	49755	15	31191	1312	8215	5808	198	78	3428
Fault Description: Fault in F3			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1089	98911	138	166	152	126	180	146	82	99
1σ	1499	98501	200	218	506	121	279	156	126	91
2σ	4125	95875	611	462	1376	123	828	200	392	131
3σ	11250	88750	1812	918	6024	87	1946	281	1047	135
5σ	48219	51781	7933	1625	29049	34	6045	209	3242	82
Fault Description: Fault in F4			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1033	98967	107	172	117	162	153	156	68	98
1σ	1545	98455	144	280	115	342	180	249	77	158
2σ	4406	95594	327	807	111	1019	262	719	127	434
3σ	12077	87923	705	2183	72	5846	328	1909	150	1084
5σ	49928	50072	1277	8602	29	30570	232	5706	77	3435

Table A.7: *Results of FDI using scalar structured residual approach with single testing - Part1*

Fault Description: Fault in F5			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1164	98836	122	179	143	128	227	182	83	100
1σ	1953	98047	137	321	259	174	541	266	120	135
2σ	6698	93302	217	1128	920	251	2714	727	352	389
3σ	20317	79683	286	3077	2635	311	10477	1689	849	993
5σ	71967	28033	151	8963	8030	176	47316	3759	1802	1770
Fault Description: Fault in F6			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1138	98862	119	160	109	152	185	219	83	111
1σ	1825	98175	228	186	125	271	274	455	144	142
2σ	6409	93591	816	291	212	1015	776	2547	347	405
3σ	19690	80310	2437	361	286	2958	1890	9949	853	956
5σ	71050	28950	7979	214	155	8909	3946	46294	1759	1794
Fault Description: Fault in F7			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1114	98886	118	158	135	143	176	164	118	102
1σ	1799	98201	180	230	234	215	284	260	240	156
2σ	6184	93816	801	678	837	617	774	673	1373	431
3σ	18642	81358	2428	1884	2451	1778	1854	1681	5651	915
5σ	69631	30369	8394	5904	8419	5770	4255	4083	31310	1496
Fault Description: Fault in F8			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1106	98894	112	184	113	151	179	168	75	124
1σ	1792	98208	168	318	186	263	270	235	111	241
2σ	6310	93690	556	1004	563	965	708	696	368	1450
3σ	19240	80760	1629	2889	1679	2735	1756	1665	812	6075
5σ	70177	29823	5418	8898	5311	8616	4065	3920	1363	32586

Table A.8: *Results of FDI using scalar structured residual approach with single testing - Part2*

Fault Description: No fault			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
N/A	1032	98968	161	161	179	159	102	85	88	97
Fault Description: Fault in F1			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1071	98929	181	178	179	186	85	103	75	104
1σ	1443	98557	311	165	235	236	118	142	129	107
2σ	3698	96302	1375	130	502	552	149	432	398	160
3σ	10039	89961	5005	77	1079	1096	184	1276	1153	168
5σ	44486	55514	28186	26	2286	2304	131	5311	5150	112
Fault Description: Fault in F2			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1079	98921	115	226	172	173	107	93	72	121
1σ	1573	98427	120	471	232	235	155	95	89	176
2σ	4423	95577	79	1964	532	577	508	147	123	493
3σ	12157	87843	40	6829	1116	1237	1444	194	141	1356
5σ	50245	49755	13	33780	2141	2339	5855	122	102	5893
Fault Description: Fault in F3			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1089	98911	139	213	196	161	104	94	92	90
1σ	1499	98501	196	281	393	142	173	96	130	88
2σ	4125	95875	451	651	1640	100	535	142	454	152
3σ	11250	88750	1057	1275	5748	53	1475	190	1284	168
5σ	48219	51781	2274	2513	31834	9	6013	114	5549	113
Fault Description: Fault in F4			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1033	98967	142	200	137	196	96	92	61	109
1σ	1545	98455	207	282	137	412	103	169	78	157
2σ	4406	95594	485	695	104	1846	144	488	155	489
3σ	12077	87923	1074	1393	58	6302	212	1440	168	1430
5σ	49928	50072	2116	2721	12	32912	131	5820	87	6129

Table A.9: *Results of FDI using scalar structured residual approach with maximized sensitivity and single testing - Part1*

Fault Description: Fault in F5			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1164	98836	136	216	217	174	125	102	79	115
1σ	1953	98047	161	[illegible]	426	213	224	134	143	206
2σ	6698	93302	373	1855	1674	386	913	315	547	635
3σ	20317	79683	561	[illegible]	5691	649	3233	572	1655	1831
5σ	71967	28033	580	24202	22624	570	12886	773	4906	5426

Fault Description: Fault in F6			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1138	98862	171	184	155	209	111	111	77	120
1σ	1825	98175	320	229	214	387	148	192	136	199
2σ	6409	93591	1465	448	386	1639	334	925	548	616
3σ	19690	80310	5283	770	639	5791	667	3128	1631	1781
5σ	71050	28950	22451	687	600	23480	817	12717	4947	5351

Fault Description: Fault in F7			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1114	98886	151	183	230	152	96	113	103	106
1σ	1799	98201	276	226	366	202	189	188	185	176
2σ	6184	93816	1337	488	1515	407	668	614	816	339
3σ	18642	81358	4774	741	5077	661	2030	1797	2829	733
5σ	69631	30369	21930	721	22531	615	5581	5274	12118	861

Fault Description: Fault in F8			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1106	98894	126	214	152	195	107	125	71	116
1σ	1792	98208	172	406	185	362	172	153	122	220
2σ	6310	93690	403	1667	384	1440	635	607	303	871
3σ	19240	80760	587	6621	595	5207	1836	1751	597	3046
5σ	70177	29823	538	23621	555	22279	5234	5007	768	12175

Table A.10: *Results of FDI using scalar structured residual approach with maximized sensitivity and single testing - Part2*

Fault Description: No fault			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
N/A	1032	98968	116	121	132	128	140	118	142	135
Fault Description: Fault in F1			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1071	98929	124	126	113	136	151	134	158	129
1σ	1443	98557	286	156	138	162	158	179	217	145
2σ	3698	96302	1592	224	297	290	199	487	417	192
3σ	10039	89961	6206	323	531	483	259	1045	945	247
5σ	44486	55514	37718	352	796	758	225	2278	2114	247
Fault Description: Fault in F2			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1079	98921	79	187	125	133	144	126	140	145
1σ	1573	98427	89	439	161	164	230	132	158	200
2σ	4423	95577	137	2217	285	331	581	171	197	504
3σ	12157	87843	191	9092	510	604	1208	217	286	1049
5σ	50245	49755	209	43510	716	826	2382	162	219	2212
Fault Description: Fault in F3			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1089	98911	86	121	171	129	156	140	158	128
1σ	1499	98501	113	161	389	135	210	145	210	136
2σ	4125	95875	206	327	2008	208	545	190	466	175
3σ	11250	88750	422	533	7294	325	1119	218	1065	274
5σ	48219	51781	679	775	41489	330	2318	182	2233	213
Fault Description: Fault in F4			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1033	98967	77	118	106	182	136	133	144	137
1σ	1545	98455	99	172	124	442	161	192	150	205
2σ	4406	95594	192	364	200	2207	223	522	226	472
3σ	12077	87923	367	639	303	9053	263	1163	280	1009
5σ	49928	50072	575	875	291	43242	192	2363	216	2174

Table A.11: *Results of FDI using structured residual vector approach with maximized sensitivity and single testing - Part1*

Fault Description: Fault in F5			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1164	98836	76	159	135	132	254	138	138	132
1σ	1953	98047	85	261	231	140	699	163	190	184
2σ	6698	93302	138	765	663	170	4028	261	340	333
3σ	20317	79683	156	1665	1422	209	15539	348	525	453
5σ	71967	28033	71	2260	1979	86	66702	172	376	321
Fault Description: Fault in F6			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1138	98862	84	122	113	158	137	225	153	146
1σ	1825	98175	157	130	142	243	173	601	195	184
2σ	6409	93591	497	183	173	693	297	3885	350	331
3σ	19690	80310	1105	234	251	1582	425	15034	555	504
5σ	71050	28950	1744	113	119	2229	207	65848	400	390
Fault Description: Fault in F7			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1114	98886	92	119	134	125	155	127	232	130
1σ	1799	98201	137	156	215	150	182	185	611	163
2σ	6184	93816	458	230	610	227	388	373	3643	255
3σ	18642	81358	1059	292	1319	291	595	594	14177	315
5σ	69631	30369	1677	130	2006	167	457	476	64540	178
Fault Description: Fault in F8			Number of samples: 100000							
Size	Fault Present?		Isolation							
	Yes	No	F1	F2	F3	F4	F5	F6	F7	F8
0.5σ	1106	98894	74	155	105	151	143	128	128	222
1σ	1792	98208	81	279	117	254	177	172	162	550
2σ	6310	93690	161	697	187	679	359	330	249	3648
3σ	19240	80760	182	1500	227	1391	559	522	300	14559
5σ	70177	29823	97	2167	109	2049	433	409	165	64748

Table A.12: *Results of FDI using structured residual vector approach with maximized sensitivity and single testing - Part2*

B

Sample Matlab programs for the EM algorithm

The following Matlab programs are used for EM-based model identification using irregularly sampled data obtained from a simulated 3rd order underdamped system.

The Matlab program for the EM algorithm is provided in Section B.1. It uses the Matlab function *Kpfs.m* (Section B.2) to obtain Kalman filter, predictor and smoother estimates and their covariances and the Matlab function *fir_mdl.m* (Section B.3) to obtain the initial model.

These Matlab programs can be used for model identification using data from other systems after a effecting a small number of changes based on the theory provided in Chapter 4.

B.1 Matlab code for EM algorithm

```
%================================
% Example - 3rd order SISO system.
%================================
clear;
clc;
% Simulation parameters
N = 5000;
Time = [1:N]';
% System definition
mu0 = zeros(3,1); % Initial state mean;
```

```
Sigma0 = zeros(3,3); % Initial state covariance
n = 3; % number of states
m = 1; % number of inputs
p = 1; % number of outputs
phi = [0.3688 0.4767 0.0114; -0.5976 0.6095 -0.5408;...
        -0.0156 -0.0686 0.0422]; % Dynamic transition matrix
gama = [0.34;0.56;0.78]; % Input transformation matrix
A = [1.2 0.96 1.5]; % Output transformation matrix
Rw = [0.0407 0.0001 0.0015; 0.0001 0.0407 -0.0020;...
        0.0015 -0.0020 0.0428]; % State noise covariance matrix
Lrw = [chol(Rw)]'; % Cholesky factor of Rw
Rv = 0.0398; % Measurement noise covariance matrix
Lrv = chol(Rv); % Cholesky factor of Rv
sys = ss(phi,gama,A,0,1);
% Simulation data
u = idinput(N,'rbs',[0 1]); % Definition of input sequence
randn('state',4854082);
w = randn(N,3)*Lrw;
randn('state',1246983);
v = randn(N,1)*Lrv;
% Simulation
% Initialization
x(1,:) = [phi*mu0]' + w(1,:);
y(1,:) = [A*[x(1,:)]' + [v(1,:)]']';
% Recursion
for t = 2:N
    x(t,:) = [phi*[x(t-1,:)]']'+ [gama*[u(t-1,:)]']' + w(t,:);
    y(t,:) = [A*[x(t,:)]' + [v(t,:)]']';
end
% Missing data
Nm = 0;
for t = 1:N
    % Drop every 3rd and 5th observation.
    if (mod(t,3)==0)|(mod(t,5)==0)
        ym(t,1) = NaN;
        Nm = Nm + 1;
    else
        ym(t,1) = y(t,1);
    end
end

%============================================================
% Model identification using the EM algorithm
%============================================================
% Initialization
[sysini]= fir_mdl(ym,u,20);
step(sys,sysini);pause(2);

phi = sysini.A;
gama = sysini.B;
A = sysini.C;
mu0     = zeros(n,1);
```

```
Sigma0 = zeros(n,n);
Rw = eye(n);
Rv = eye(p);
% Recursion
for niter = 1:30
    %=======================================================
    % State estimation - Kalman Predictor, Filter, Smoother
    % and Lag-one covariance smoother
    %=======================================================
    [Xp,Pp,Yp,Xf0,Pf0,Xf,Pf,Yf,K,Xs0,Ps0,J0,Xs,Ps,Ys,J,...
            PsL1] = Kpfs(mu0,Sigma0,phi,gama,Rw,A,Rv,n,m,p,ym,u,N);
    %=======================================================
    % Calculation of Negative Log Likelihood (NLL) function.
    % Used to monitor the convergence of EM algorithm.
    %=======================================================
    % Initialization
    Nve_log_lkhd = 0;
    % Recursion
    for t = 1:N
        if ~isnan(ym(t,:))
            % Innovation
            epsilon(t,:) = ym(t,:) - Yp(t,:);
            % Covariance of innovation
            SigmaK{t} = A*Pp{t}*A' + Rv;
            % Contribution of current sample to NLL function
            Of_t = p*log(2*pi) + log(det(SigmaK{t})) + epsilon(t,:)*...
                (SigmaK{t}\eye(size(SigmaK{t})))*[epsilon(t,:)]';
        else
            Of_t = 0;
        end
        Nve_log_lkhd = Nve_log_lkhd + Of_t;
        clear Of_t;
    end
    Lk(niter,1) = Nve_log_lkhd;
    sprintf('Iter␣No:␣␣␣%d␣␣Objfn:␣␣%0.2f',niter-1,Nve_log_lkhd)
    %======================================================================
    % Calculation of data-dependent regression matrices - alpha1 to alpha6
    %======================================================================
    %============================================
    alpha1 = zeros(n,n);
    for t = 1:N
        Temp1 = Xs(t,:)'*Xs(t,:) + Ps{t};
        alpha1 = alpha1 + Temp1;
        clear Temp1;
    end
    %============================================
    alpha2 = Xs(1,:)'*Xs0' + PsL1{1};
    for t = 2:N
        Temp2 = Xs(t,:)'*Xs(t-1,:) + PsL1{t};
        alpha2 = alpha2 + Temp2;
        clear Temp2;
    end
```

```
%========================================
alpha3 = Xs0*Xs0' + Ps0;
for t = 2:N
    Temp3 = Xs(t-1,:)'*Xs(t-1,:) + Ps{t-1};
    alpha3 = alpha3 + Temp3;
    clear Temp3;
end
%========================================
alpha4 = zeros(n,m);
for t = 2:N
    Temp4 = Xs(t,:)'*u(t-1,:);
    alpha4 = alpha4 + Temp4;
    clear Temp4;
end
%========================================
alpha5 = zeros(n,m);
for t = 2:N
    Temp5 = Xs(t-1,:)'*u(t-1,:);
    alpha5 = alpha5 + Temp5;
    clear Temp5;
end
%========================================
alpha6 = zeros(m,m);
for t = 2:N
    Temp6 = u(t-1,:)'*u(t-1,:);
    alpha6 = alpha6 + Temp6;
    clear Temp6;
end
%========================================
PSI = [alpha2 alpha4];
GAMA1 = [alpha3 alpha5; alpha5' alpha6];
%=====================================================================
% Calculation of data-dependent regression matrices - beta1 to beta4
%=====================================================================
beta1 = 0;
for t = 1:N
    if ~isnan(ym(t,:))
        Temp1 = ym(t,:)'*ym(t,:);
        beta1 = beta1 + Temp1;
        clear Temp1;
    end
end
%========================================
beta2 = 0;
for t = 1:N
    if ~isnan(ym(t,:))
        Temp2 = ym(t,:)'*Xs(t,:);
        beta2 = beta2 + Temp2;
        clear Temp2;
    end
end
%========================================
    beta3 = 0;
    for t = 1:N
        if ~isnan(ym(t,:))
            Temp3 = Xs(t,:)' * Xs(t,:) + Ps{t};
            beta3 = beta3 + Temp3;
            clear Temp3;
        end
    end
    %==========================================
    beta4 = Nm*Rv;
    %==========================================
    % New model parameters
    %==========================================
    mu0 = Xs0;
    Sigma0 = zeros(n,n); % Fixed at 0
    phigama = PSI*(GAMA1\eye(size(GAMA1)));
    phi = phigama(1:n,1:n);
    gama = phigama(1:n,n+1:n+m);
    Rw = (alpha1 - PSI*(GAMA1\eye(size(GAMA1)))*PSI')/N;
    A = beta2*(beta3\eye(size(beta3)));
    Rv = (beta4 + beta1 - beta2*(beta3\eye(size(beta3)))*beta2')/N;
    %=========================================================
    sysf = ss(phi,gama,A,0,1);
    step(sys,sysf); pause(2);
end
```

B.2 Matlab code for Kalman Smoother

```
function [Xp,Pp,Yp,Xf0,Pf0,Xf,Pf,Yf,K,Xs0,Ps0,J0,Xs,Ps,Ys,J,...
              PsL1] = Kpfs(mu0,Sigma0,phi,gama,Rw,A,Rv,n,m,p,ym,u,N);
%
% Function used for state estimation.
% System Description
% Multivariate Dynamic system with manipulated inputs.
% Represented by nth order discrete-time state-space model
%
%  State equation:   x(t) = phi * x(t-1) + gama * u(t-1) + w(t)
%  Output equation:  y(t) =  A  * x(t)   + v(t)
%
% Usage:
%         [Xp,Pp,Yp,Xf0,Pf0,Xf,Pf,Yf,K,Xs0,Ps0,J0,Xs,Ps,Ys,J,...
%             PsL1] = Kpfs(mu0,Sigma0,phi,gama,Rw,A,Rv,n,m,p,ym,u,N);
%
% Inputs:
%
mu0, Sigma0: Initial state mean and covariance: (n x 1), (n x n).
%          phi: Dynamic transition matrix: (n x n).
%         gama: Input transformation matrix: (n x m).
%            A: Output transformation matrix: (p x n).
%           Rw: State noise covariance: (n x n).
%           Rv: Measurement noise covariance: (p x p).
%            n: Number of states: (1 x 1).
%       m: Number of inputs: (1 x 1).
%       p: number of outputs: (1 x 1).
%       ym: Output data matrix: (N x p).
%       u: Input data matrix: (N x m).
%       N: Number of samples: (1 x 1).
%
% Outputs:
%       Xp: Predicted states: (N x n).
%       Pp: Predicted state covariance: N cells. (n x n).
%       Yp: Predicted outputs: (N x p).
%       Xf0: Filtered estimate of initial state: (n x 1).
%       Pf0: Filtered estimate of initial state covariance: (n x n).
%       Xf: Filtered states: (N x n).
%       Pf: Filtered state covariance: N cells, (n x n).
%       Yf: Filtered outputs: (N x p).
%       K: Kalman Gain: N cells, (n x p).
%       Xs0: Smoothed estimate of initial state: (n x 1).
%       Ps0: Smoothed estimate of initial state covariance: (n x n).
%       J0: Initial Update gain in smoother: (n x n).
%       Xs: Smoothed states: (N x n).
%       Ps: Smoothed state covariance: N cells, (n x n).
%       Ys: Smoothed outputs: (N x p).
%       J: Update gain in smoother: (N-1) cells, (n x n).
%       PsL1: Lag-one covariance smoother: N cells, (n x n).

%=====================================================================
% State estimation - Kalman Predictor. Filter and Smoother.
%=====================================================================
% Kalman Predictor and Filter.
%=====================================================================
% Arrangement of results
%=====================================================================
% Kalman Predictor
%=====================================================================
% Predicted state: Xp(t,:)
% Predicted state covariance: Pp{t}
% Predicted output: Yp(t,:)
%=====================================================================
% Kalman Filter
%=====================================================================
% Filtered state: Xf(t,:)
% Filtered state covariance: Pf{t}
% Filtered output: Yf(t,:)
%=====================================================================
% Kalman gain
%=====================================================================
% Update gain: K{t}
%=====================================================================
%=====================================================================
% Initialization
%=====================================================================
Xf0 = mu0;
```

```
Pf0 = Sigma0;
% Prediction
Xp(1,:) = [phi*Xf0]';
Pp{1} = phi*Pf0*phi'+Rw;
Yp(1,:) = [A*[Xp(1,:)]']';
% Update gain
Temp = (A*Pp{1}*A'+Rv);
Temp = Temp\eye(size(Temp)); % Matrix inversion
K{1} = Pp{1}*A'*Temp;
clear Temp;
% Filtering
Xf(1,:) = [[Xp(1,:)]'+K{1}*([ym(1,:)]'-A*[Xp(1,:)]')]';
Pf{1} = (eye(n)-K{1}*A)*Pp{1};
Yf(1,:) = [A*[Xf(1,:)]']';
%======================================================================
%======================================================================
% Recursion
%======================================================================
for t = 2:N
    if ~isnan(ym(t,:))
        % yt is observed
        % Prediction
        Xp(t,:) = [phi*[Xf(t-1,:)]' + gama*[u(t-1,:)]']';
        Pp{t} = phi*Pf{t-1}*phi'+Rw;
        Yp(t,:) = [A*[Xp(t,:)]']';
        % Update gain
        Temp = (A*Pp{t}*A'+Rv);
        Temp = Temp\eye(size(Temp)); % Matrix inversion
        K{t} = Pp{t}*A'*Temp;
        clear Temp;
        % Filtering
        Xf(t,:) = [[Xp(t,:)]'+K{t}*([ym(t,:)]'-A*[Xp(t,:)]')]';
        Pf{t} = (eye(n)-K{t}*A)*Pp{t};
        Yf(t,:) = [A*[Xf(t,:)]']';
    else
        % yt is missing
        % Prediction
        Xp(t,:) = [phi*[Xf(t-1,:)]']' + [gama*[u(t-1,:)]']';
        Pp{t} = phi*Pf{t-1}*phi'+Rw;
        Yp(t,:) = [A*[Xp(t,:)]']';
        % Update gain
        K{t} = zeros(n,p);
        % Filtering
        Xf(t,:) = Xp(t,:);
        Pf{t} = Pp{t};
        Yf(t,:) = [A*[Xf(t,:)]']';
    end
end
%======================================================================
%======================================================================
% Kalman Smoother and Lag-one covariance smoother
%======================================================================
% Arrangement of results
%==================================================================
% Kalman Smoother
%==================================================================
% Smoothed state: Xs(t,:)
% Smoothed state covariance: Ps{t}
% Smoothed output: Ys(t,:)
%==================================================================
% Kalman Smoother Gain
%==================================================================
% Update Gain: J{t}
%==================================================================
% Lag-one Covariance Smoother
%==================================================================
% Lag-one covariance: PsL1{t}
%==================================================================
%==================================================================
% Initialization
%==================================================================
Xs(N,:) = Xf(N,:);
Ps{N} = Pf{N};
Ys(N,:) = [A*[Xs(N,:)]']';
%==================================================================
% Recursion
%==================================================================
for t = N:-1:2
    J{t-1} = Pf{t-1}*phi'*[Pp{t}\eye(size(Pp{t}))];
    Xs(t-1,:) = [[Xf(t-1,:)]' + J{t-1}*([Xs(t,:) - Xp(t,:)]')]';
    Ps{t-1} = Pf{t-1} + J{t-1}*(Ps{t}-Pp{t})*J{t-1}';
    Ys(t-1,:) = [A*[Xs(t-1,:)]']';
end
%==================================================================
% Final element
%==================================================================
J0 = Pf0*phi'*[Pp{1}\eye(size(Pp{1}))];
Xs0 = Xf0 + J0*([Xs(1,:) - Xp(1,:)]');
Ps0 = Pf0 + J0*(Ps{1} - Pp{1})*J0';
%==================================================================
% Initialization
%==================================================================
PsL1{N} = (eye(n) - K{N}*A)*phi*Pf{N-1};
%==================================================================
% Recursion
%==================================================================
for t = N-1:-1:2
    PsL1{t} = Pf{t}*J{t-1}' + J{t}*(PsL1{t+1} - phi*Pf{t})*J{t-1}';
end
%==================================================================
% Final element
%==================================================================
PsL1{1} = Pf{1}*J0' + J{1}*(PsL1{2} - phi*Pf{1})*J0';
```

B.3 Matlab code for Initial FIR model

```
function sysini = fir_mdl(y,u,nf);
%
% This function estimates an initial state-space model based on
% FIR coefficients using kung's svd technique.
%
% Usage: sysini = fir_mdl(y,u,nf)
%
% Output: sysini - Initial state space model.
%
% Inputs: y - System output data matrix.
%             ('Nxm' matrix where 'N' is number of samples and
%              'm' is number of outputs).
%         u - Manipulated input data.
%             ('Nxp' matrix where 'N' is number of samples and
%              'p' is number of inputs).
%         nf - Number of FIR coefficients to be used.
%         FIR Model of the form H(z) = h{0} + h{1} z^-1 + ...
%              + h{nf} z^-nf, will be used.
%
% This function can handle missing data in the output
% if the missing samples are replaced by NaN's

N = size(y,1);
ny = size(y,2);
nu = size(u,2);

% nf should be an odd number for easy formation of the hankel matrix
if mod(nf,2)==0
    nf = nf+1;
end
es = (nf+1)/2;

% There will be {nu x ny x (nf+1)} FIR coefficients

F = {};

% Find the least squares estimate for the FIR coefficients

for i = 0:nf
    for j = 1:ny
        for k = 1:nu
            Y1 = y(i+1:N,j);
            U1 = u(1:N-i,k);

            % Clean these matrices
            sz = size(Y1,1);
            Y2 = [];
            U2 = [];
            for el = 1:sz
                if ~isnan(Y1(el,1))
```

```
                    Y2 = [Y2;Y1(el,1)];
                    U2 = [U2;U1(el,1)];
                end
            end
            theta = pinv(U2)*Y2;

            Ftemp(j,k) = theta;

            % clear the used variables
            clear theta Y2 U2 sz el U1 Y1;
        end
    end
    F{i+1} = Ftemp;
    clear Ftemp;
end

% The state-space matrices

% 'D' matrix can be read out as the first FIR coefficient matrix
% or the FIR coefficient matrix at zero lag.
D_est = zeros(size(F{1}));

% Assume for the time being that the indices go from 0 to 2s-1.
%
% Then the hankel matrix is:
%   Hc = [ F{1} F{2}   . . . F{s}
%          F{2} F{3}   . . . F{s+1}
%           .    .     . . .  .
%          F{s} F{s+1} . . . F{2s-1} ]
%
% However, the indices go from 1 to 2s with the first index
% corresponding to the lag zero impulse response coefficient.
%
% Hence the hankel matrix is:
%   Hc = [ F{2}   F{3}   . . . F{s+1}
%          F{3}   F{4}   . . . F{s+2}
%           .      .     . . .  .
%          F{s+1} F{s+2} . . . F{2s} ]

Hc = [];
% Form the Hc matrix
for i = 1:es
    Htemp = [];
    for j = 1:es
        Htemp = [Htemp;F{i+j}];
    end
    Hc = [Hc Htemp];
    clear Htemp;
end

% Perform SVD on the Hc matrix

```

```
[U11,S11,V11] = svd(Hc);

% Choosing the order
plot(diag(S11),'*-'); xlabel('State-space␣dimension');
ylabel('Singular␣value');
title('Singular␣values␣of␣Markov␣parameter␣Hankel␣matrix');

en = input('Enter␣the␣desired␣system␣order:␣');


% Truncate the SVD according to the chosen order
U11 = U11(:,1:en);
S11 = S11(1:en,1:en);
V11 = V11(:,1:en);



Sval = diag(S11);
for i = 1:size(Sval,1)
    Sval(i,1) = real(sqrt(Sval(i,1)));
end
S11r = diag(Sval);

% Extended observability matrix
Gamma_s = U11*S11r;

% Extended controllability matrix
Omega_s = S11r*V11';


% 'B' and 'C' matrices are obtained from first block column
% and first block row of the extended controllability
% and extended observability matrices respectively.

B_est = conj(Omega_s(:,1:nu));
C_est = conj(Gamma_s(1:ny,:));

% 'A' matrix is obtained from the shift invariant structure
% of the extended observability or extended controllability
% matrix.
%
% Observe that:
% 1. Gamma_s(ny+1:end,:) = Gamma_s(1:end-ny,:)*A
% 2. Omega_s(:,nu+1:end) = A*Omega_s(:,1:end-nu)

A_est1 = conj(Gamma_s(1:end-ny,:)\Gamma_s(ny+1:end,:));
A_est2 = [conj(Omega_s(:,1:end-nu)'\Omega_s(:,nu+1:end)')]';
A_est = (A_est1 + A_est2)/2;

sysini = ss(A_est,B_est,C_est,D_est,1);
```

Bibliography

Afonso, P., J. Ferreira and J. Castro (1998). Sensor fault detection and identification in a pilot plant under process control. *Trans IChemE* **76**(Part A), 490–498.

Amirthalingam, R., S.W. Sung and J.H. Lee (2000). A Two Step Procedure for Data-Based Modeling for Inferential Predictive Control System Design. *AIChE Journal* **46**, 1974–1988.

Ansley, C.F. and R. Kohn (1983). Exact Likelihood of Vector Autoregressive-Moving Average Process with Missing or Aggregated Data. *Biometrika* **70**(1), 275–278.

Åström, K.J. (1980). Maximum likelihood and prediction error methods. *Automatica* **16**(5), 551–574.

Basseville, M. and I.V. Nikiforov (1993). *Detection of Abrupt Changes-Theory and Applications*. Prentice-Hall, Englewood Cliffs, NJ.

Bauer, D. (2001). Order estimation for subspace methods. *Automatica* **37**, 1561–1573.

Chan, N.N. and T.K. Mak (1983). Estimation of Multivariate Linear Functional Relationships. *Biometrika* **70**(1), 263–267.

Chen, T. and B. Francis (1995). *Optimal Sampled-Data Control Systems*. Springer-Verlag.

Chou, C.T. and M. Verhaegen (1997). Subspace Algorithms for the Identification of Multivariable Dynamic Errors-in-Variables Models. *Automatica* **33**(10), 1857–1869.

Chou, C.T., R. Johansson and M. Verhaegen (1999). Continuous-time identification of SISO systems using laguerre functions. *IEEE Trans. Signal Processing* **47**(2), 349–362.

Chow, E.Y. and A.S. Willsky (1984). Analytical redundancy and the design of robust failure detection systems. *IEEE Trans. Auto. Cont.* **AC-29**, 603–614.

Deistler, M. (2000). Model Identification and Adaptive Control. In: *System Identification - General Aspects and Structure*. Springer-Verlag.

Dempster, A.P., N.M. Laird and D.B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B* **39**, 1–38.

Denn, M.M. and R. Lavie (1982). Dynamics of plants with recycle. *Chem. Eng. J. (London)* **24**, 54–59.

Ergon, R. (1998). Dynamic system multivariate calibration. *Chemometrics and Intelligent Laboratory Systems* **44**, 135–146.

Forssell, U. and L. Ljung (1999). Closed-loop identification revisited. *Automatica* **35**, 1215–1241.

Frank, P. and S.X. Ding (1997). Survey of robust residual generation and evaluation methods in observer-based fault detection systems. *Journal of Process Control* **7**(6), 403–424.

Frank, P.M. (1990). Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy - a survey and some new results. *Automatica* **26**, 459–474.

Freidland, B. (1961). Sampled-data control systems containing periodically varying members. In: *Proceedings of the 1961 IFAC World Congress*. Vol. 37. pp. 361–367.

Fuller, W.A. (1987). *Measurement Error Models*. John Wiley & Sons, Inc., New York.

Ge, W. and C.Z. Fang (1988). Detection of faulty components via robust observation. *Int. J. Control* **47**, 581–599.

Gertler, J. (1998). *Fault Detection and Diagnosis in Engineering Systems*. Marcel Dekker.

Gertler, J. and D. Singer (1990). A New Structural Framework for Parity Equation based Failure Detection and Isolation. *Automatica* **26**, 381–388.

Gertler, J., W. Li, Y. Huang and T.J. McAvoy (1999). Isolation-enhanced principal component analysis. *AICHE Journal* **45**(2), 323–334.

Gibson, S. and B. Ninness (2000). On the relationship between state-space-subspace-based and maximum-likelihood system identification methods. In: *Proceedings of the 2000 IEEE International Conference on Decision and Control*. Sydney, Australia.

Gleser, L.J. (1981). Estimation in a Multivariate "Errors in Variables" Regression Model: Large Sample Results. *The Annals of Statistics* **9**, 24–44.

Gleser, L.J. and J.T. Hwang (1987). The nonexistense of $100(1-\alpha)\%$ confidence sets of finite expected diameter in errors-in-variables and related models. *The Annals of Statistics* **15**(4), 1351–1362.

Gopaluni, R.B., H. Raghavan, R.S. Patwardhan and S.L. Shah (2003). On the identification of consistent models for processes with material and energy recycle. Presented at AIChE Annual Meeting, Nov. 2003, San Francisco, USA.

Gupta, A.K. (1952). Estimation of the Mean and Standard Deviation of a Normal population from a Censored sample. *Biometrika* **39**, 260–273.

Gupta, N.K. and R.K Mehra (1974). Computational aspects of maximum likelihood estimation and reduction in sensitivity function calculations. *IEEE Trans. Auto. Cont.* **19**, 774–783.

Huang, B. and S.L. Shah (1997). Closed loop identification: A two step approach. *J. Proc. Cont.* **7**(6), 425–438.

Inc, OSI Software (2002). Pi data storage component overview. Technical report. Retrieved Dec 17th 2002, from http:// www.osisoft.com/ 270.htm, 2002.

Isaksson, A.J. (1993). Identification of ARX-models subject to missing data. *IEEE Trans. Auto. Cont.* **38**, 813–819.

Isermann, R. (1984). Process Fault Detection Based on Modeling and Estimation Methods - a Survey. *Automatica* **20**, 387–404.

Jackson, J.E. and G.S. Mudholkar (1979). Control Procedures for Residuals Associated With Principal Component Analysis. *Technometrics* **21**(3), 341–349.

Johansson, K.H. (2000). The quadruple-tank process: A multivariable laboratory process with an adjustable zero. *IEEE Trans. Cont. Sys. Tech.* **8**(3), 456–465.

Johansson, R. (1994). Identification of continuous-time methods. *IEEE Trans. Signal Processing* **42**(4), 887–897.

Johansson, R., M. Verhaegen and C.T. Chou (1999). Stochastic Theory of Continuous-Time State-Space Identification. *IEEE Trans. Signal Processing* **47**(1), 41–51.

Jones, R.H. (1980). Maximum likelihood fitting of ARMA models to time series with missing observations. *Technometrics* **22**, 389–395.

Kailath, T. (1980). *Linear Systems*. Prentice-Hall. Englewood Cliffs, New Jersey.

Khargonekar, P.P., K. Poolla and A. Tannenbaum (1985). Robust control of linear time-invariant plants using periodic compensation. *IEEE Trans. Auto. Cont.* **30**, 1088–1096.

Kowalczuk, Z. and J. Kozłowski (2000). Continuous-time approaches to identification of continuous-time systems. *Automatica* **36**, 1229–1236.

Kranc, G.M. (1957). Input-output analysis of multirate feedback systems. *IEE Trans. Auto. Control* **3**, 21–28.

Kresta, J.V., J.F. MacGregor and T.E. Marlin (1991). Multivariate Statistical Monitoring of Processes. *Can. J. Chem. Eng.* **69**(1), 35–47.

Kresta, J.V., T.E. Marlin and J.F. MacGregor (1994). Development of Inferential Process Models using PLS. *Computers Chem. Engng.* **18**, 597–611.

Ku, W., R.H. Storer and C. Georgakis (1995). Disturbance Detection and Isolation by Dynamic Principal Component Analysis. *Chemometrics and Intelligent Laboratory Systems* **30**, 179–196.

Kung, S.Y. (1978). A new identification and model reduction algorithm via singular value decomposition. In: *12th Asilomar Conference on Circuits, Systems and Computers*. Pacific Grove, CA. pp. 705–714.

Kwok, K.E., M. ChongPing and G.A. Dumont (2001). Seasonal Model-Based Control of Processes with Recycle Dynamics. *Ind. Eng. Chem. Res.* **40**, 1633–1640.

Lakshminarayanan, S. and H. Takada (2001). Empirical modelling and control of processes with recycle: some insights via case studies. *Chem. Eng. Sci.* **56**(11), 3327–3340.

Lakshminarayanan, S., S.L. Shah and K. Nandakumar (1997). Modelling and Control of Multivariable Processes: The Dynamic Projection to Latent Structures Approach. *AIChE Journal* **43**, 2307–2323.

Larsson, E.G. and E.K. Larsson (2002). The Cramer-Rao bound for continuous-time AR parameter estimation with irregular sampling. *Circuits, Systems and Signal Processing* **21**(6), 581–601.

Larsson, E.G. and E.K. Larsson (2004). The CRB for Parameter Estimation in Irregularly Sampled Continuous-Time ARMA Systems. *IEEE Signal Processing Letters* **11**(2), 197–200.

Lewis, F.L. (1986). *Optimal Estimation*. John Wiley and Sons. New York.

Li, D., S.L. Shah and T. Chen (2001*a*). Identification of fast-rate models from multirate data. *International Journal of Control* **74**(7), 680–689.

Li, D., S.L. Shah and T. Chen (2002). Analysis of dual-rate inferential control systems. *Automatica* **38**(6), 1053–1059.

Li, W. and S.L. Shah (2002). Structured residual vector-based approach to sensor fault detection and isolation. *Journal of Process Control* **12**(3), 429 – 443.

Li, W., H. Raghavan and S.L. Shah (2001*b*). Subspace identification of residual models for FDI in continuous-time systems. In: *Proc. of CHEMFAS-4*. Jejudo Island, Korea. pp. 224–229.

Li, W., H. Raghavan and S.L. Shah (2003). Subspace identification of continuous time models for process fault detection and isolation. *J. Proc. Cont.* **13**(5), 407–421.

Little, R.J.A. and D.B. Rubin (1987). *Statistical Analysis with Missing Data*. John Wiley and Sons.

Liu, B. and J. Si (1997). Fault isolation filter design for time-invariant systems. *IEEE Trans. Auto. Cont.* **42**, 704–707.

Ljung, L. (1999). *System Identification: Theory for the User*. Prentice-Hall, Inc.. Englewood Cliffs, New Jersey. Second Edition.

Ljung, L. and T. Glad (1994). *Modeling of Dynamic Systems*. Prentice Hall, Englewood Cliffs, NJ.

Luyben, W.L. (1994). Snowball effects in reactor-separator processes with recycle. *Ind. Eng. Chem. Res.* **33**(2), 299–305.

Marlin, T.E. (1995). *Process Control: Designing Processes and Control Systems for Dynamic Performance*. McGraw-Hill Inc.

Massoumnia, M. (1986). A geometric approach to the synthesis of failure detection filter. *IEEE Trans. Auto. Cont.* **31**, 1867–1888.

Middleton, R.H. and G.C. Goodwin (1990). *Digital Control and Estimation*. Prentice-Hall. Englewood Cliffs, New Jersey.

Moonen, M., B. DeMoor, L. Vandenberghe and J. Vandewalle (1989). On and off-line identification of linear state-space models. *International Journal of Control* **49**, 219–232.

Morud, J.C. and S. Skogestad (1994). Effects of recycle on dynamics and control of chemical processing plants. *Comput. Chem. Engng.* **18**, 529–534.

Narasimhan, S. and S.L. Shah (2003). Model identification and error covariance matrix estimation from noisy data using PCA. In: *IFAC ADCHEM'03*. Hong Kong.

Ninness, B. and S. Gibson (2002). Robust and simple algorithms for maximum likelihood estimation of multivariable systems. In: *Proceedings of the 2002 IFAC World Congress*. Barcelona.

Overschee, P.V. and B. DeMoor (1996). *Subspace Identification for Linear Systems-Theory, Implementation, Applications*. Kluwer Academic Publisher, Boston, MA.

Parrish, J.R. and C.B. Brosilow (1985). Inferential Control Algorithms. *Automatica* **21**(5), 527–538.

Pearson, R.K. and B.A. Ogunnaike (1996). Nonlinear model identification. In: *Nonlinear Process Control* (M.A. Henson and D.E. Seborg, Eds.). Prentice-Hall. Englewood Cliffs, N.J.

Qin, S.J. (1993). Partial least squares regression for recursive system identification. In: *Proceedings of the 32nd Conference on Decision and Control.*

Qin, S.J. and T.J. McAvoy (1992). Nonlinear PLS modeling using neural networks. *Comput. Chem. Eng.* **16**(4), 379–391.

Qin, S.J. and W. Li (1999). Detection, Identification, and Reconstruction of Faulty Sensors with Maximized Sensitivity. *AIChE Journal* **45**(9), 1963–1976.

Qin, S.J. and W. Li (2001). Detection and identification of faulty sensors in dynamic processes. *AIChE Journal* **47**, 1581 – 1593.

Raghavan, H., S.L. Shah, R. Kadali and B. Doucette (2003). Application of PLS-based regression for monitoring bitumen recovery in a separation cell. In: *IFAC ADCHEM'03*. Hong Kong.

Rao, C.R. (2001). *Linear Statistical Inference and its Applications*. John Wiley and Sons. New York. Second Edition.

Ricker, N.L. (1988). The use of biased least-squares estimators for parameters in discrete-time pulse response models. *Ind. Eng. Chem. Res.* **27**, 343–350.

Romagnoli, J.A. and G. Stephanopoulos (1981). Rectification of Process Measurement Data in the Presence of Gross Errors. *Chem. Eng. Sci.* **36**(11), 1849–1863.

Russell, E.L., L.H. Chiang and R.D. Braatz (2000). *Data-driven techniques for Fault Detection and Diagnosis in Chemical Processes*. Springer-Verlag.

Sagara, S. and Z.Y. Zhao (1990). Numerical integration approach to on-line identification of continuous-time systems. *Automatica* **26**(1), 63–74.

Schweppe, F.C. (1965). Evaluation of likelihood functions for gaussian signals. *IEEE Trans. Inform. Theo.* **IT-4**, 294–305.

Shumway, R.H. and D.S. Stoffer (1982). An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis* **3**(4), 253–264.

Shumway, R.H. and D.S. Stoffer (2000). *Time Series Analysis and Its Applications*. Springer-Verlag. New York.

Singhal, A. and D.E. Seborg (2000). Dynamic Data Rectification Using the Expectation Maximization Algorithm. *AIChE Journal.* **46**(8), 1556–1565.

Sinha, N.K. and G.P. Rao (1991). *Identification of continuous time systems*. Kluwer Academic Publishers.

Söderström and P. Stoica, T. (1983). *Instrumental Variable Methods for System Identification*. Springer-Verlag. Lectures Notes in Control and Information Sciences.

Söderström, T. (1999). On the cramer-rao lower bound for estimating continuous-time autoregressive parameters. In: *Proceedings of the 1999 IFAC World Congress*. Beijing.

Söderström, T., H. Fan, B. Carlsson and S. Bigi (1997). Least squares parameter estimation of continuous time ARX models from discrete time data. *IEEE Trans. Auto. Cont.* **42**, 659–673.

Sorenson, H.W., Ed.) (1985). *Kalman Filtering: Theory and Application*. IEEE PRESS. A volume in the IEEE PRESS Selected Reprints Series.

Stoica, P. and T. Söderström (1998). Partial least squares: a first-order analysis. *Scandinavian Journal of Statistics* **25**, 17–24.

Tanaka, M. and T. Katayama (1990). Robust Identification and Smoothing for Linear System with Outliers and Missing Data. In: *Proceedings of the 1990 IFAC World Congress*. Tallinn.

Thornhill, N.F., M.A.A.S. Choudhury and S.L. Shah (2004). The Impact of Compression of Data-Driven Process Analysis. *Journal of Process Control* **14**(4), 389–398.

Titterington, D. (1984). Recursive parameter estimation using incomplete data. *Journal of the Royal Statistical Society, Series B* **46**(2), 257–267.

Unbehauen, H. and G.P. Rao (1990). Continuous-time Approaches to System Identification-A Survey. *Automatica* **26**(1), 23–35.

Verhaegen, M. (1994). Identification of the deterministic part of MIMO state space models given in innovations form from input-output data. *Automatica* **30**(1), 61–74.

Verhaegen, M. and P. Dewilde (1992*a*). Subspace model identification. Part I: the output-error state-space model identification class of algorithms. *International Journal of Control* **56**(5), 1187–1210.

Verhaegen, M. and P. Dewilde (1992*b*). Subspace model identification. Part II: Analysis of the elementary output-error state-space model identification algorithm. *International Journal of Control* **56**(5), 1211–1241.

Viberg, M. (1995). Subspace-based methods for the identification of linear time-invariant systems. *Automatica* **31**(12), 1835–1853.

Wang, J., T. Chen and B. Huang (2004). Multirate sampled-data systems: computing fast-rate models. *Journal of Process Control* **14**(1), 79–88.

Wang, L. and P. Gawthrop (2001). On the estimation of continuous transfer functions. *Int. J. Control* **74**, 889–904.

Wentzell, P.D., D.T. Andrews, D.C. Hamilton, K. Faber and B.R. Kowalski (1997). Maximum likelihood Principal Component Analysis. *Journal of Chemometrics* **11**, 339–366.

White, J.E. and J.L. Speyer (1987). Detection filter design: Spectral theory and algorithms. *IEEE Trans. Auto. Cont.* **32**, 593–603.

Whitfield, A.H. and N. Messali (1987). Integral-equation approach to system identification. *Int. J. Control* **45**(4), 1431–1445.

Willsky, A.S. (1976). A survey of design methods for failure detection in dynamic systems. *Automatica* **12**, 601–611.

Wise, B.M. and N.B. Gallagher (1996). The Process Chemometrics Approach to Process Monitoring and Fault Detection. *Int. J. Control* **6**(6), 329–348.

Wu, C.F. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics* **11**, 95–103.

Young, P.C. (1981). Parameter estimation for continuous-time models. *Automatica* **17**(1), 23–39.

Zhang, P., S.X. Ding, G.Z. Wang and D.H. Zhou (2003). A frequency domain approach to fault detection in sampled-data systems. *Automatica* **39**(7), 1303–1307.